Escola d'Enginyeria de Telecomunicació i Aeroespacial de Castelldefels
UNIVERSITAT POLITÈCNICA DE CATALUNYA

TRABAJO FINAL DE GRADO

TÍTULO DEL TFG: Análisis del sistema de comunicaciones BLE

TITULACIÓN: Grado en Ingeniería de Sistemas de Telecomunicación

AUTOR: Jorge Alcalá Colmenero

DIRECTOR: Jose Luis Valenzuela

FECHA: 21 de julio del 2017

Título: Análisis del sistema de comunicaciones Bluetooth Low Energy

Autor: Jorge Alcalá Colmenero

Director: José Luis Valenzuela

Fecha: 21 de julio del 2017

Resumen

Este trabajo pretende analizar un sistema de comunicaciones Bluetooth Low Energy (BLE) completo. Para ello se ha hecho un estudio teórico y se han realizado diferentes experimentos con el objetivo de obtener unos resultados y saber cuál es la posibilidad de introducir la tecnología en el mundo de las cursas o eventos deportivos (p. ej. Maratones, cursas de ciclistas, etc). Con este proyecto se busca dar una mayor fiabilidad en cuanto a tiempos y resultados obtenidos mediante chips de gran durabilidad.

Las características más importantes que se han tenido en cuenta para la propuesta han sido el alcance de los dispositivos programados con varias configuraciones, el consumo del dispositivo utilizado en dichas configuraciones, los inconvenientes que uno de los dispositivos esté en movimiento y como esto afecta en la comunicación y el tiempo de detección de una gran cantidad de dispositivos, como sería en el caso de una cursa, entre otras.

La estructura del trabajo se compone por una parte teórica, en la cual se introduce al mundo del Bluetooth, explicando primero el funcionamiento de BLE, junto con su protocolo. También se hace una comparación con los otros estándares para ver la evolución de esta tecnología a lo largo del tiempo y cual son las novedades que han ido introduciendo. A continuación, se estudian los dos procedimientos más importantes para la comunicación, procedimientos de *Scanning*/*Advertising*, que a partir de estos procesos se puede conseguir establecer una conexión.

En la parte experimental se han realizado diferentes pruebas, con la intención de caracterizar el dispositivo utilizado y ver cuál es su comportamiento. Para ello ha sido necesario estudiar los dispositivos utilizados, tanto transmisor como receptor, con qué parámetros se configuran, cómo se modifican y configuran mediante comandos HCI. El resto de pruebas han servido para saber si alguna de las posibles configuraciones es viable para llevar a cabo el objetivo de esta tesis, que es ser capaces de detectar a miles de personas mediante chips que implementen el estándar de bajo consumo.

Title: Analysis of the Bluetooth Low Energy Communications System

Author: Jorge Alcalá Colmenero

Director: José Luis Valenzuela

Date: July 21st 2017

Overview

This thesis aims to analyse a complete Bluetooth Low Energy (BLE) communications system. A theoretical study and different experiments have been carried out in order to evaluate if Bluetooth technology can be efficiently deployed in sports events (e.g. marathon, cycle races, etc). To do so, high durability chips are used, providing a higher reliability in terms of time and obtained results.

The main features that have been considered to develop the proposed configuration have been: the range and the consumption of the devices by using different configurations, the way in which the movement affects the communication between nodes and the time needed to discover a huge amount of Bluetooth devices.

The structure of this paper is composed by an introduction, where an overview of Bluetooth technology is presented, focusing on BLE's performance and its protocol. A study is made throughout the different versions of this specification in order to figure out the updates introduced in each release. Furthermore, the two most important stages for the communication (*Scanning* / *Advertising*) are described. They allow to establish a communication between several devices.

Regarding the experimental part different tests have been conducted, aiming to characterise the used devices and define its behaviour. For that purpose transmitters and receivers are analysed by means of checking different parameters and how they can be modified trough HCI commands. The rest of tests determined if any of the potential configurations is suitable to cover the aim of this thesis, providing an effective Bluetooth scheme to properly detect devices in highly populated environments (in order of thousand devices).

ÍNDICE

INTRODUCCIÓN 10

CAPÍTULO 1. DESCRIPCIÓN DE BLUETOOTH LOW ENERGY 12

1.1 Arquitectura 12
1.1.1 Capa Física 14
1.1.2 Capa de Enlace 16

1.2 Evolución de Bluetooth 17
1.2.1 Versión 1.0 y actualizaciones 19
1.2.2 Versión 2.0 y 2.1 + EDR 19
1.2.3 Versión 3.0 + HS 20
1.2.4 Compatibilidad de las versiones 20

CAPÍTULO 2. DESCRIPCIÓN DEL PROCESO DE ADVERTISING 22

2.1 Intervalo de Advertising 23

2.2 Advertising Channel PDU 24

2.3 Tipos de Advertisings 25
2.3.1 Connectable Undirected Advertising 25
2.3.2 Connectable Directed Advertising 26
2.3.3 Nonconnectable Undirected Advertising 27
2.3.4 Scannable Undirected Advertising 28

CAPÍTULO 3. DESCRIPCIÓN DEL PROCESO DE SCANNING 29

3.1 Passive Scanning 30

3.2 Active Scanning 31

CAPÍTULO 4: PRUEBAS CON DISPOSITIVOS BLE 32

4.1 Caracterización Dispositivo BLE 32
4.1.1 Caracterización en Vertical 33
4.1.2 Caracterización en horizontal 34

4.2 Caracterización de Energía 34
4.2.1 Consumo configuración a 100ms 35
4.2.2 Consumo configuración 300ms y 500ms 36

4.3 Test Cobertura 37

4.4 Test Advertising Scannable 39
4.4.1 Dispositivo BELKIN 39
4.4.2 Dispositivo TRUST 41

4.5 Test con dispositivos en movimiento 43
4.5.1 Test 4 Dispositivos 43
4.5.2 Test 52 Dispositivos 45

4.6 Test Discovery Time 48

4.6.1 Discovery Time Nonconnectable Undirected Advertising 49
4.6.2 Discovery Time Scannable Undirected Advertising 50

4.7 Test Potencia en función del Canal 51
4.7.1 Test en entorno aislado 51
4.7.2 Test con influencia de dispositivos y personas 53

4.8 Análisis de la Maratón de Barcelona 55
4.8.1 Análisis a 5 km de la Maratón de Barcelona 55
4.8.2 Análisis a 10 km de la Maratón de Barcelona 57

CAPÍTULO 5: CONCLUSIONES Y RESULTADOS 59

BIBLIOGRAFÍA 61

ANEXOS 63

Anexo 1: Scripts 63
SCAN ACTIVO 63
SCAN PASIVO 63
SCAN CONTINUO 64
SCAN 50% 64
Habilitar el Advertiser 65

ÍNDICE DE FIGURAS

Figura 1: Protocolo BLE 13
Figura 2: Comunicación Bluetooth 14
Figura 3: Distribución Canales BLE..... 15
Figura 4: Estados y roles de los dispositivos..... 16
Figura 5: Clasificación de Dispositivos por Alcance 18
Figura 6: Clasificación según la capacidad 18
Figura 7: Compatibilidad entre versiones Bluetooth 21
Figura 8: Modo Broadcast 22
Figura 9: Intervalo de Advertising..... 23
Figura 10: Advertising *Channel* PDU..... 24
Figura 11: Tipos de Advertisings 25
Figura 12: Connectable Undirected Advertising 25
Figura 13: Flujo de mensajes (Connectable Undirected Advertising) 26
Figura 14: Connectable Directed Advertising 27
Figura 15: Flujo de mensajes (Connectable Directed)..... 27
Figura 16: Nonconnectable Advertising..... 28
Figura 17: Scannable Undirected Advertising..... 28
Figura 18: Tiempo de escaneo..... 29
Figura 19: Parámetros para configurar el Scanning 30
Figura 20: Dispositivo Red Bear BLE Nano..... 32
Figura 21: Puntos para la caracterización del dispositivo..... 33
Figura 22: Diagrama radiación BLE Red Bear Nano (pos. Vertical) 33
Figura 23: Diagrama radiación BLE Red Bear Nano (pos. Horizontal)..... 34
Figura 24: Consumo paquete 26 Bytes (100ms) 35
Figura 25: Consumo paquete 10 Bytes (100ms) 35
Figura 26: Consumo paquete 1 Byte (100ms)..... 36
Figura 27: Consumo diferentes tamaños de paquete a 300ms 36
Figura 28: Consumo diferentes tamaños de paquete a 500ms 37
Figura 29: Datos Test Cobertura 37
Figura 30: Evolución de la potencia en función de la distancia 38
Figura 31: Evolución de la potencia en función de la distancia 38
Figura 32: Comunicación Belkin ADV_DISCOVERABLE Canal 38..... 39
Figura 33: Comunicación Belkin ADV_DISCOVERABLE (Efecto Captura)..... 40
Figura 34: Comunicación Trust pérdida paquete..... 41
Figura 35: Comunicación Trust colisión canal 37 42
Figura 36: RSSI Primera pasada en movimiento..... 43
Figura 37: Potencia recibida para cada pasada 44
Figura 38: Primera distancia Test Cobertura 1ª configuración 45
Figura 39: Distancia máxima Test Cobertura 1ª configuración..... 46
Figura 40: Distancia máxima en movimiento 1ª Configuración..... 46
Figura 41: Distribución dispositivos Test movimiento 47
Figura 42: Discovery Time ADV_NONCONN_IND..... 49
Figura 43: Tiempo medio descubrimiento ADV_NONCONN_IND..... 49
Figura 44: Discovery Time ADV_SCAN_IND 50
Figura 45: Tiempo medio de descubrimiento ADVSCAN_IND 50
Figura 46: Frecuencia de potencias en entorno aislado (Canal 37) 51
Figura 47: Frecuencia de potencias en entorno aislado (Canal 38) 52

Figura 48: Frecuencia de potencias en entorno aislado (Canal 39) 52
Figura 49: Frecuencia de potencias en entorno interferente (Canal 37)..... 53
Figura 50: Frecuencia de potencias en entorno interferente (Canal 38)..... 53
Figura 51: Frecuencia de potencias en entorno interferente (Canal 39)..... 54
Figura 52: Máximo Nº corredores por segundo en zona cobertura [5 Km]..... 56
Figura 53: Máximo Nº corredores en zona de cobertura [5 Km]..... 56
Figura 54: Máximo Nº corredores por segundo en zona cobertura [10km]..... 57
Figura 55: Máximo Nº corredores en zona de cobertura [10 Km]..... 58
Figura 56: Nº corredores por segundo en Salida..... 58

ÍNDICE DE ACRÓNIMOS

ADV	Advertising
ADV_DIRECT_IND	Connectable Directed Advertising
ADV_IND	Connectable Undirected Advertising
ADV_NONCONN_IND	Non-Connectable Undirected Advertising
ADV_SCAN_IND	Scannable Undirected Advertising
AFH	Adaptative Frequency-Hopping
ATT	Attribute Protocol
BLE	Bluetooth Low Energy
EDR	Enhanced Data Rate
EIR	Extended Inquiry Response
GAP	Generic Access Profile
GATT	Generic Attribute Profile
GFSK	Gaussian Frequency Shift Keying
HS	High Speed
ISM	Industrial Scientific Medical
IoT	Internet of Things
PDU	Packet Data Unit
PHY	Physical
PSK	Phase Shift Keying
RSSI	Receibed Signal Strength Indicator
SCAN_REQ	Scan Request
SCAN_RSP	Scan Response
SIG	Special Interest Group
TX	Transmisor
WLAN	Wireless Local Area Network
WPAN	Personal Area Network

INTRODUCCIÓN

El principal objetivo de este proyecto es analizar un sistema de comunicaciones BLE, con la intención de averiguar si es viable conseguir detectar a miles de corredores durante eventos deportivos, mediante dispositivos Bluetooth de bajo consumo. Para ello se han realizado diferentes pruebas, con el objetivo de saber cuál es la configuración idónea para programar los dispositivos, teniendo en cuenta todos los posibles efectos externos que nos pueden afectar a la hora de hacer las medidas.

El trabajo está dividido en cinco capítulos claramente estructurados. Se empieza describiendo el funcionamiento básico de Bluetooth, así como la evolución de este a lo largo del tiempo, incluyendo las novedades que se han ido añadiendo en cada versión.

Los dos siguientes capítulos se centran en explicar los dos procesos más importantes a la hora de comunicarse dos dispositivos Bluetooth, proceso de *Advertising* y proceso de *Scanning*, mediante los cuales los dispositivos se dan a conocer y escuchan el medio respectivamente. Hay diferentes tipos de Advertisings y por ello se adjuntan capturas del análisis obtenidas a partir del osciloscopio con la intención de decidir qué tipo es el más adecuado para cumplir el objetivo.

La parte experimental se basa en experimentos y análisis de datos, con la finalidad de corroborar si la propuesta para la detección de corredores será útil y viable. Para ello se ha caracterizado el dispositivo BLE utilizado y se han hecho diferentes pruebas y medidas a lo largo del proyecto, obteniendo los resultados mediante gráficas y tablas.

Uno de los principales problemas es determinar el radio de cobertura máximo que tenemos que abarcar. Para conocer esta distancia a la cual se puede situar un corredor de nuestro receptor se ha llevado a cabo el análisis de la maratón de Barcelona, haciendo un seguimiento de las calles y tramos por los que esta pasa. Teniendo en cuenta que los corredores llevan una velocidad de aproximadamente 25 km/h, el alcance de los dispositivos que están enviando mensajes de *Advertising*, se reduce considerablemente si lo comparamos con una configuración estática. Para ello, la configuración del dispositivo tendrá que estar pensada teniendo en cuenta este factor y otros factores que pueden influir en la pérdida de paquetes.

Otro de los factores que también puede afectar a esto, es que se transmitan los paquetes por tres canales diferentes. Es por eso que se ha realizado un test en función de los 3 canales por los que se transmiten los mensajes, para observar cuál es el comportamiento de cada uno y cómo esto puede influir en el resultado final.

Para este tipo de eventos dónde encontramos cantidades masivas de gente, hay que tener en cuenta que en un cierto momento de la cursa, pueden llegar muchos corredores de golpe a nuestra zona de cobertura, motivo para saturar los canales con paquetes de *Advertisings*, de tal manera que el dispositivo receptor no podría llegar a ser capaz de detectar a todos los dispositivos. Para intentar satisfacer esta necesidad, se han realizado pruebas de tiempos de descubrimiento con varios dispositivos transmitiendo a la vez, simulando este caso real. Las pruebas se han realizado para dos tipos de Advertising diferentes y a partir de los resultados obtenidos, se decidirá cuál será el tipo utilizado en nuestra configuración.

Una vez realizadas todas estas pruebas y después de haber analizado todas las capturas, observando los resultados en mano, se dedica un capítulo a las conclusiones obtenidas, para saber si se puede llevar a cabo la propuesta de este proyecto.

CAPÍTULO 1. DESCRIPCIÓN DE BLUETOOTH LOW ENERGY

Bluetooth Low Energy (BLE) se trata de una de las últimas funcionalidades añadidas al estándar Bluetooth, después de que saliera la cuarta versión. Consideramos que BLE es un estándar que comparte bastantes similitudes con su versión 4.0, ya que ha sido creado con un objetivo diferente. Esta versión se creó para abarcar la necesidad de un muy bajo consumo que Classic Bluetooth no es capaz de cumplir. El principal objetivo es que cualquier dispositivo u objeto que se conecte mediante este sistema, sea capaz de tener una vida útil de meses o incluso años, siendo alimentado por una batería.

BLE no está pensado para el *streaming* de datos, ya que su capacidad por segundo es baja, sino que ha sido diseñado para la transmisión de pequeñas cantidades de información en los cuales se aplican tiempos de conexión pequeños, la cual cosa reduce el consumo de energía. De esta forma, conseguimos que los dispositivos estén activos solo cuando se proceda a la transmisión de datos.

Teniendo en cuenta esto, podemos llegar a conseguir aplicaciones muy útiles que solo trabajen cuando queramos y podamos obtener datos de tiempos, imprescindibles para clasificaciones de cursas o cualquier deporte que necesite una supervisión de resultados.

1.1 Arquitectura

Este tipo de redes pretenden principalmente la transferencia de información a distancias cortas entre un grupo privado o no de dispositivos mediante una red inalámbrica de área personal, es decir, no requieren una infraestructura muy elaborada. El objetivo es conseguir redes ad hoc de bajo coste y consumo, optimizando así una máxima durabilidad.

La topología de red de éste estándar es de tipo estrella, en la cual se distinguen dos roles. Por un lado tenemos el dispositivo central o Master, los cuales están escuchando el canal radio para saber que dispositivos BLE están disponibles y son capaces de establecer conexiones en la capa de enlace hacia otros dispositivos periféricos (Slaves).

Los dispositivos periféricos adquieren el rol de esclavo y solo son capaces de tener una conexión hacia un dispositivo central, también es capaz de emitir mensajes para anunciar su presencia en varios modos. Puede enviar datos en modo Broadcast, eventos de *advertising*, con la intención de no esperar ninguna conexión, esto permite enviar información a los dispositivos que se encuentren en el estado de Scanning sin la necesidad de tener que establecer una conexión Master-Slave.

Para poder llevar a cabo las conexiones, SIG de Bluetooth define una pila de protocolos que se encargan de la gestión de los dispositivos, conexiones y la interfaz de las aplicaciones. Esta pila de protocolo se divide en tres partes: *Controller*, *Host* y *Applications*.

El Controller es el dispositivo físico en sí. Este dispositivo permite enviar y recibir señales radio, así como es capaz de interpretarlas mediante paquetes de información que le llegan. Esta parte del protocolo contiene la Capa Física, la Capa de Enlace y la interfaz de control de host.

Por otro lado, el Host es la pila de software. Se encarga de administrar las comunicaciones entre dos o más dispositivos que pretendan intercambiar información entre sí. Esta segunda parte del protocolo contiene la Capa de control de enlace lógico y de protocolo de adaptación, el administrador de seguridad, protocolo de atributo (ATT), el perfil de atributo genérico (GATT) y el perfil de acceso genérico (GAP).

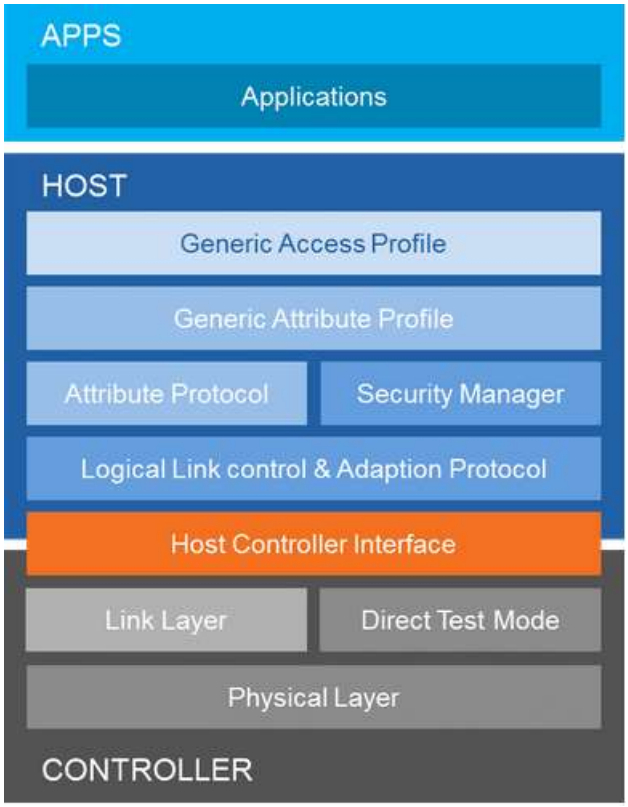

Figura 1: Protocolo BLE

Dependiendo del uso que se le vaya a dar, las aplicaciones utilizan la pila software y mediante ella se accede al controlador para llevar a cabo la tarea deseada. Las capas de Enlace y Física que son las que más importancia tienen a la hora de las pruebas realizadas, son las que se describen a continuación.

1.1.1 Capa Física

Esta capa es la encargada de realizar las transformaciones que se le aplican a las secuencias de bits que pretenden ser transmitidos hacia un receptor situado en un lugar concreto. Los bits de estas secuencias se manipulan desde un ordenador asignándole a cada uno un nivel eléctrico. Por ejemplo, puede decirse que encontramos un bit “1” cuando vemos un cierto nivel de voltaje o corriente y un bit “0” cuando su nivel es cero o nulo. Cuando realizamos una transmisión de bits siempre se hace una transformación del tipo de señal, de tal manera que el receptor sea capaz de entender y recuperar la secuencia de bits enviados.

Esta tecnología tiene en común muchos aspectos con la versión clásica de Bluetooth. Ambas utilizan la misma banda de frecuencia de 2.4 GHz y también comparten el tipo de modulación. Utilizan una GFSK a 1Mbps, pero con la diferencia del índice de modulación que no es el mismo para ambos casos. Sin embargo EDR (Enhanced Data Rate), la segunda versión de Bluetooth, utiliza una combinación entre GFSK y PSK. Entre otras diferencias respecto a esta capa tenemos que BLE utiliza casi la mitad de canales respecto a su primera versión y el ancho de los canales también varía, la cual cosa hace que estas dos versiones sean incompatibles y no se puedan comunicar entre sí. Esto se puede solucionar incluyendo en los dispositivos el *Dual Mode*, que es la implementación de ambas tecnologías pero conmutando los parámetros de modulación y los canales por los cuál transmito.

Figura 2: Comunicación Bluetooth

La capa física es la encargada de enviar las señales al aire, transmitiendo y recibiendo los bits mediantes ondas. Se utiliza la banda de frecuencia desde los 2302 MHz hasta los 2480 MHz, frecuencias que pertenecen a la banda ISM (Industrial Scientific Medical). Éstas han sido reservadas internacionalmente para un uso no comercial, adquiriendo un mayor protagonismo en las redes WLAN (Wifi) o WPAN (Bluetooth).

BLE utiliza un total de 40 canales (numerados del 0 al 39), los cuales están separados 2 MHz como podemos observar en la figura 3. De todos estos canales, encontramos 3 que son dedicados para los mensajes de *Advertising* (canales 37, 38 y 39). Gracias a estos paquetes enviados por estos tres canales conseguimos que los dispositivos se den a conocer, es decir, den a conocer a otros dispositivos que están disponibles para establecer una conexión o simplemente enviar información sin la necesidad de llegar a establecer la conexión. El resto de canales son utilizados para la transmisión de datos.

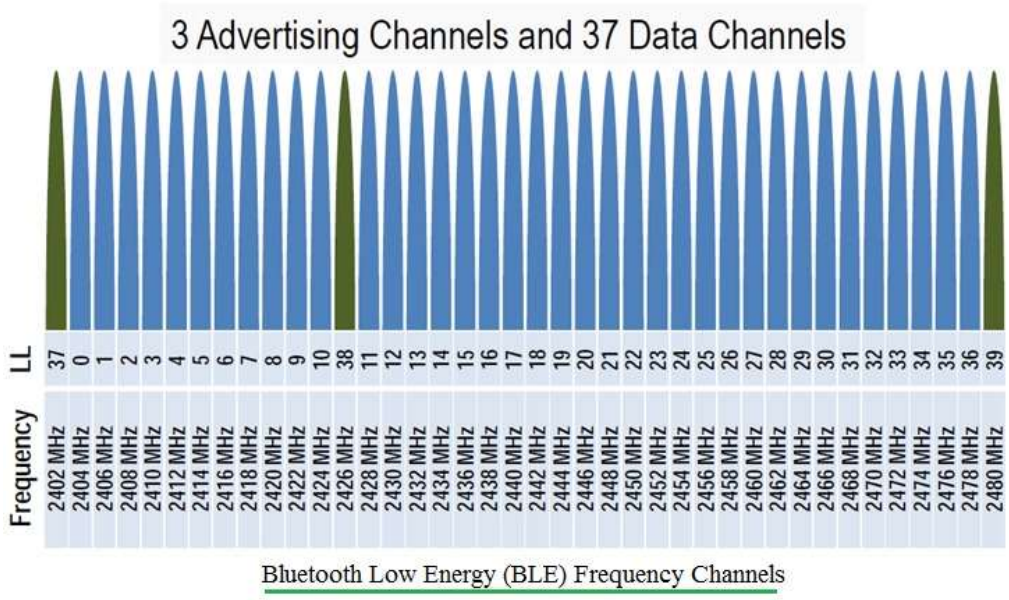

Figura 3: Distribución Canales BLE

Los tres canales señalados en verde en la figura 3 son en los que se envían los paquetes de *Advertising*. Están situados de una forma conveniente, ya que esta distribución está pensada para evitar interferencias que pueden ser causadas por cualquier otro dispositivo que puede coexistir en la misma banda de frecuencias, ya que se trata de un trozo de espectro abierto y que cualquier persona puede utilizar, siempre cumpliendo unas ciertas normas de transmisión. Cuando se realiza una conexión, a parte que ya se ha tenido en cuenta la separación entre los canales, también incorpora la técnica de FHSS (Frequency Hopping Spread Spectrum) que también pretende reducir las interferencias en nuestros dispositivos. Se trata de una técnica de modulación en espectro ensanchado, en el que la señal va saltando de frecuencia en frecuencia sincrónicamente con el transmisor, consiguiendo una serie de radiofrecuencias alternas. De este modo, los receptores no deseados en la comunicación, recibirán señales inteligibles. Una señal de la cual solo conseguirían recuperar unos pocos bits, innecesarios para descifrar el mensaje.

Con esta técnica conseguimos hasta tres ventajas que ayudan a mejorar nuestro sistema. Entre ellas, hacer que nuestras señales sean más resistentes al ruido y a la interferencia, que nuestra señal sea más difícil de interceptar y conseguimos compartir la banda de frecuencia entre muchos tipos de transmisores con una interferencia mínima.

1.1.2 Capa de Enlace

Esta capa es la que se encarga de las conexiones entre dispositivos y de la estructura de los paquetes. Recibe las peticiones de la capa de red y utiliza los servicios de la capa física. El objetivo de esta capa es que la información fluya, sin errores cuando se establece una comunicación. Encontramos varios estados en los que un dispositivo puede encontrarse.

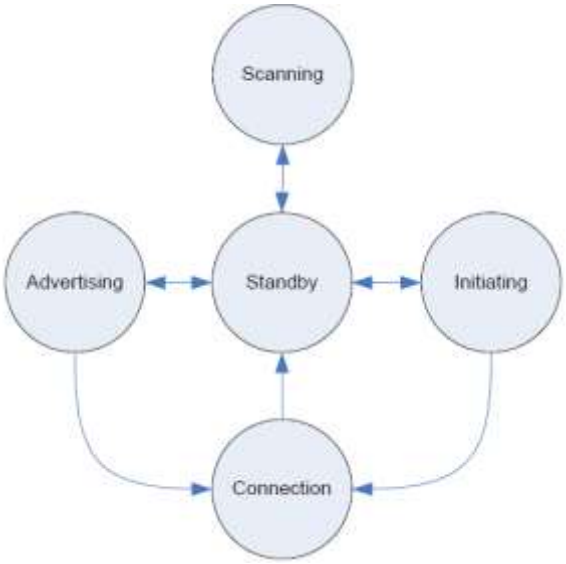

Figura 4: Estados y roles de los dispositivos

- **Standby**: Estado de descanso. El dispositivo no transmite ni recibe información. Los dispositivos entran en este estado cuando no van a establecer una conexión, de manera que ahorran energía.

- **Scanning**: El dispositivo se encuentra escaneando el canal continuamente. En este estado se escuchan los paquetes de *advertising* que se envían por los 3 canales y se utiliza para explorar otros dispositivos y saber si están activos.

- **Advertising**: El dispositivo con el rol de *Advertiser* es el encargado de enviar paquetes de *Advertising* por los 3 canales correspondientes. Su principal objetivo es darse a conocer y que el resto de dispositivos sepan que está activo y reciban los paquetes que éste envía. En este estado el dispositivo también escucha cualquier posible respuesta de algún dispositivo que haya recibido su mensaje de *Advertising*.

- **Initiating**: Es el estado por el cual pasa el dispositivo central previo al estado conectado. En este estado el dispositivo recibe los mensajes de *Advertising* de otros dispositivos, pero hasta que no recibe el mensaje del dispositivo deseado, el Master no envía los datos correctos para la conexión.

- **Connected**: Se ha establecido la conexión entre el dispositivo Slave y el Master. Se puede proceder al intercambio de paquetes o finalizar la conexión.

Para el dispositivo que hace el rol de Slave, cuando se encuentra en estado de *Advertising*, se puede considerar también como su estado inicial antes de que se establezca la conexión. El estado conectado es el último, en el cuál se pueden intercambiar información los dos dispositivos. La información se intercambia mediante eventos de conexión que están predefinidos de forma periódica, es decir, los parámetros se han de configurar previamente a la conexión.

A cualquier dispositivo que se encuentre en el estado de escaneo se le llama *Scanner* y cuando se inicia la conexión, *Initiator*. El dispositivo que utiliza los canales de *Advertising* definidos anteriormente, se le denomina *Advertiser*. Los canales de datos no se utilizan hasta que no se ha establecido la conexión.

El Scanner es el dispositivo que está escaneando continuamente los canales de *Advertising* para ver si cualquier otro dispositivo los está utilizando para establecer una conexión o enviar información. Existen varios tipos de *Advertising*, pero para este trabajo sólo hemos utilizado cuatro tipos.

- **Connectable Undirected**: El dispositivo que se encuentra en estado de escaneo puede conectarse o sólo recibir los mensajes de *Advertising*.

- **Connectable Directed**: Dirigido a un dispositivo Bluetooth concreto.

- **Nonconnectable**: Solo transmite información, no se establece una conexión con otro dispositivo.

- **Scannable Undirected**: Es exactamente el mismo que el anterior, pero con este tipo de *Advertising* se puede solicitar información extra.

Esta es solo una breve descripción de los cuatro tipos de *Advertising* que tenemos. Más adelante se hará una descripción completa de cada uno, explicando los parámetros más importantes que tienen.

1.2 Evolución de Bluetooth

Bluetooth Low Energy es la cuarta versión existente que ha lanzado Bluetooth Special Interest Group, que se encargan de agrupar las principales compañías de computación, telecomunicaciones y dispositivos electrónicos. Como cualquier tecnología, nace de una idea poco elaborada que es lanzada al mercado, pero poco a poco se va mejorando conforme van apareciendo las nuevas versiones.

Encontramos varias clases de dispositivos Bluetooth que se clasifican de la siguiente manera:

- Clase 1: Rango de hasta 100m, potencia de consumo de 100 mW.
- Clase 2: Rango de hasta 20m, potencia de consumo de 2.5 mW.
- Clase 3: Rango menor a 1m, potencia de consumo de 1 mW.

Esta es una forma de clasificar los diferentes dispositivos Bluetooth , según su alcance y potencia consumida (ver [1]). Estos no son parámetros fijos y dependen del dispositivo utilizado y los componentes que lo dotan. Los valores comentados son aproximaciones de cómo se comportan los dispositivos, ya que estos valores puedan variar en función del entorno. Depende de sí encontramos obstáculos o medios físicos, interferencias con otros dispositivos inalámbricos trabajando a la misma frecuencia o el nivel de batería del dispositivo.

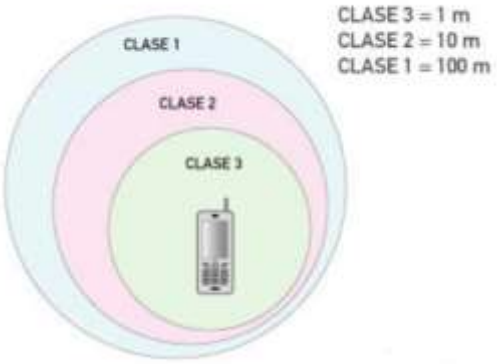

Figura 5: Clasificación de Dispositivos por Alcance

Los dispositivos que llevan incorporado este protocolo se pueden comunicar entre sí, siempre y cuándo se encuentran dentro de su alcance. Las comunicaciones se establecen mediante radiofrecuencia, de tal manera que no hace falta que los dispositivos estén enfrentados o alineados entre sí, se pueden encontrar en habitaciones separadas si la potencia de transmisión es suficiente para que el otro dispositivo lo pueda detectar.

Estos dispositivos también se pueden clasificar según la capacidad del canal, ancho de banda que ocupan.

Versión	Ancho de banda (BW)
Versión 1	1 Mbit/s
Versión 2.0 + EDR	3 Mbit/s
Versión 3.0 + HS	24 Mbit/s
Versión 4.0 (BLE)	24 Mbit/s

Figura 6: Clasificación según la capacidad

1.2.1 Versión 1.0 y actualizaciones

La primera versión de Bluetooth se utilizó para la transmisión de datos, pero actualmente se encuentra en desuso, ya que generó muchos problemas de comunicación entre los dispositivos. Constaba mucho conseguir que dos dispositivos pudieran interoperar entre sí.

Las actualizaciones de esta primera versión consiguieron la ratificación como estándar IEEE 802.15.1 [2]. También consiguieron conexiones más rápidas y detección de errores a los dispositivos. Se añadió soporte para los canales no cifrados y un indicador de señal recibida (RSSI).

Una mejora importante es el salto de frecuencia adaptable de espectro ampliado (AFH), que se trata de una variación del FHSS. Se basa en utilizar solo las “buenas” frecuencias, evitando los canales “malos” que sufren un desvanecimiento selectivo, porque algún tercero está tratando de comunicarse en la misma banda o directamente porque la banda está saturada. La mejora de velocidad también es otra característica a comentar, entre otras mejoras que se realizaron, consiguiendo tasas de hasta 1 Mbit/s.

1.2.2 Versión 2.0 y 2.1 + EDR

Una de las principales características de esta versión fue la facilidad que se consiguió para establecer comunicaciones. Para ello se creó un menú dónde un dispositivo podía agregar a otro dispositivo, para detectar y conectarse automáticamente con el otro.

Pero la más importante es la introducción de una velocidad de datos mejorada, EDR. Esto permitió acelerar la transferencia de datos, llevando a esta versión hasta una tasa de 3 Mbit/s. Se consiguió gracias a una combinación de las modulaciones GFSK y PSK con dos variantes a π/4-DQPSK y 8DPSK.

Entre otras a destacar encontramos una mejora de la seguridad y la incorporación de EIR, que se trata de una respuesta amplia de investigación. Esta puede llegar a ser muy útil, ya que durante la comunicación se puede proporcionar más información para permitir un mejor filtrado de los dispositivos antes de que se llegue a establecer la conexión.

1.2.3 Versión 3.0 + HS

La tercera versión de Bluetooth que incorpora HS, soporta velocidades de transferencia de datos de hasta 24 Mbit/s entre sí. La conexión Bluetooth se utiliza para la negociación y el establecimiento, mientras que los datos de alta velocidad se realizan mediante un enlace 802.11.

Su principal cambio se trata de AMP (Alternative MAC/PHY), que es la adición del estándar anterior comentado como transporte de alta velocidad. En esta versión la transmisión en HS no es obligatoria y por lo tanto hay dispositivos marcados con “+ HS” que incorporan el enlace. Cualquier otro dispositivo que no tenga esa marca, solo acepta una característica introducida en esta versión.

La alternativa AMP se utiliza cuando se tienen que enviar grandes cantidades de datos, como por ejemplo canciones o vídeos. Se utiliza PHY MAC 802.11 para el transporte de datos, sto conlleva que cualquier sistema que incorpore esta versión solo entrará en modo de bajo consumo cuando el sistema esté inactivo.

1.2.4 Compatibilidad de las versiones

Una vez explicadas todas las versiones pueden surgir ciertas dudas. Una de las más importantes es la compatibilidad que puede haber entre las diferentes clases de dispositivos. Es decir, ¿Un dispositivo de una clase es compatible con otro dispositivo de una clase diferente? La respuesta es sí, es posible la interacción entre dispositivos de diferente clase sin problemas, siempre y cuando los dispositivos que pretenden interactuar se ajusten al dispositivo con la clase más alta, debido al rango de alcance. Ver figura 5.

La segunda clasificación que hemos hecho ha sido respecto a las versiones Bluetooth. Existe compatibilidad entre diferentes versiones Bluetooth? Teniendo en cuenta hasta la versión 4, ésta es compatible con cualquier versión anterior siempre que la última versión no sea del tipo Smart Device, ya que usa protocolos de comunicación distintos a los empleados por el Bluetooth clásico.

Dentro de la versión 4.0, encontramos dos versiones renombradas debido a que no todos los dispositivos con esta versión tienen el mismo protocolo. La versión Bluetooth Smart sólo es compatible con dispositivos BLE (Bluetooth Smart Ready). Y la versión Bluetooth Smart Ready (versión 4.2 [3]) puede interactuar con dispositivos que incorporan el Bluetooth clásico como con dispositivos BLE. En la siguiente tabla se muestra la compatibilidad existente entre diferentes versiones.

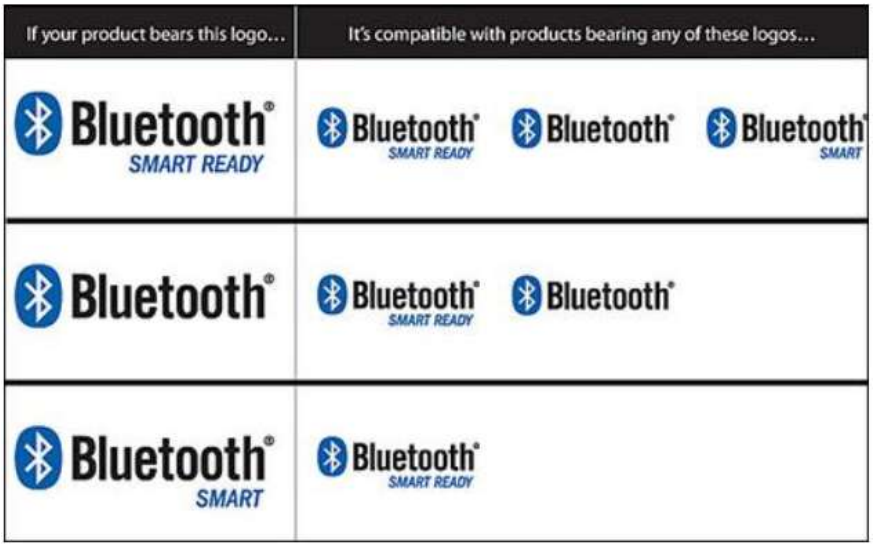

Figura 7: Compatibilidad entre versiones Bluetooth

CAPÍTULO 2. DESCRIPCIÓN DEL PROCESO DE ADVERTISING

En este capítulo se va a hacer una descripción detallada de todos los tipos de mensajes de *Advertising* que hay, como funcionan y la utilidad que puede llegar a tener cada uno de ellos. Hasta ahora hemos definido dos formas para intercambiar información entre dispositivos BLE. Una de ellas sería mediante mensajes de *Advertising* y Scan responses (respuesta que envía el dispositivo que está escaneando) y la otra mediante conexiones establecidas entre dos o varios dispositivos.

Cuando se establece una conexión entre varios dispositivos, éstos pueden intercambiar información entre sí, sin ningún tipo de restricción. Los paquetes que se envían no serán captados por otros dispositivos, ya que éstos van dirigidos a un dispositivo BLE en concreto. Los procedimientos que se requieren en una conexión son mucho más complejos comparado con otros métodos.

Los paquetes de *Advertising* llevan campos de información a transmitir, pero el dispositivo central (Master) puede solicitar información extra mediante un Scan Request. En estos métodos no hay seguridad alguna, ya que este modo de difusión amplia se trata de Broadcast, dónde un dispositivo envía información y ésta puede ser captada por muchos receptores de manera simultánea sin tener que estableces una clave o descifrar el mensaje. Por lo tanto, la información que se transmite en esta banda es pública y cualquiera con un dispositivo receptor está preparado para recibirla.

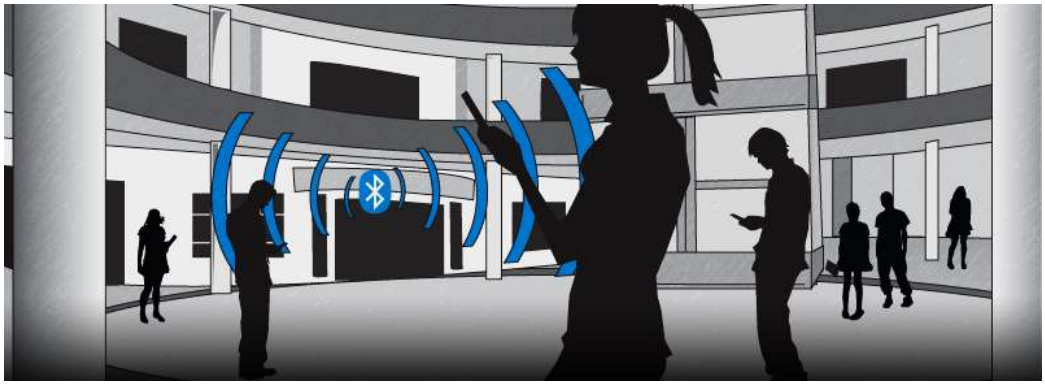

Figura 8: Modo Broadcast

En la Figura 8 observamos como varios dispositivos periféricos se comunican con un dispositivo central que es el que está mandando la información y el resto la están recibiendo. Es un ejemplo del modo Broadcast.

2.1 Intervalo de Advertising

El intervalo de *Advertising* es un parámetro que tienen en común los cuatro tipos de *Advertising* comentados en el capítulo 1. Cuando el dispositivo se encuentra en estado de *Advertising*, se envían periódicamente los paquetes por los tres canales reservados para ello. La separación temporal que hay entre ellos se compone por dos intervalos, un intervalo fijo (AdvInterval) y un retardo aleatorio (AdvDelay).

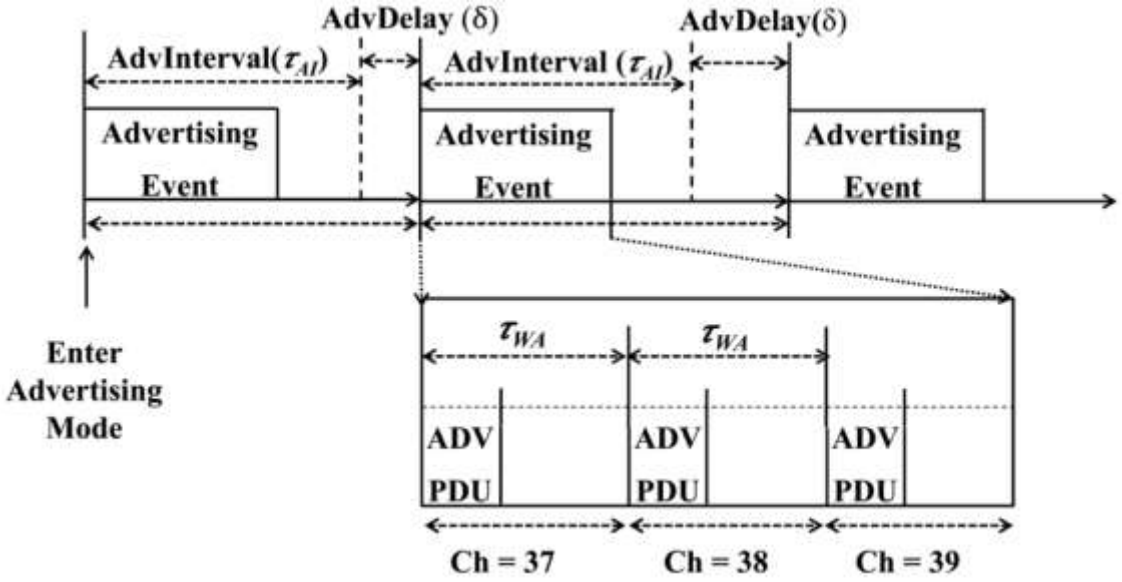

Figura 9: Intervalo de Advertising

El intervalo fijo se trata de un parámetro configurable en el dispositivo y puede variar entre 20ms y 10’24s, en saltos de 0’625ms. Dependiendo la información que queramos transmitir lo configuramos con el tiempo deseado. Para los *Advertisings* Nonconnectable y Scannable existe una excepción y el tiempo mínimo del intervalo fijo ha de ser 100ms.

El retardo aleatorio se trata de un tiempo que se añade después de cada evento de *advertising* (posterior al intervalo fijo). La duración de este retardo es variable entre 0ms y 10ms y se escoge de forma aleatoria. Gracias a este retardo añadido se consigue reducir el número de colisiones entre paquetes de *Advertising*, también BLE consigue mejorar la robustez del protocolo, consiguiendo que el Scanner sea capaz de detectar más fácilmente los *Advertisings* enviados.

2.2 Advertising Channel PDU

La estructura del paquete de *Advertising* es la misma que la de datos. Cada paquete de *Advertising* está compuesto por un preámbulo, una dirección de acceso, un PDU (Packet Data Unit) de hasta 39 bytes, dónde se enviará la información y un CRC de 3 Bytes. En la figura 10 se observa la estructura del paquete.

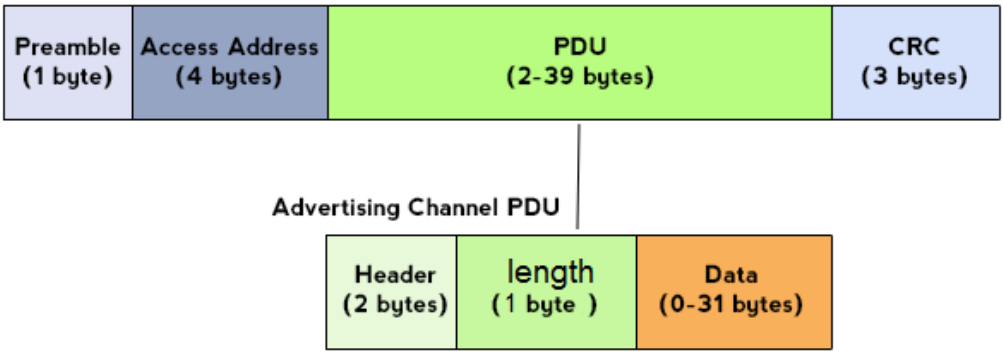

Figura 10: Advertising *Channel* PDU

Cada campo del paquete realiza su función:

- **Preámbulo**: Se trata de un patrón fijo de 8 bits. Siempre sigue la estructura 10101010 o 01010101.

- **Dirección de Acceso**: Encontramos dos tipos de direcciones de acceso, las de los *Advertisings* y las direcciones de acceso de datos.

- **Cabecera**: Este campo se encarga de indicar el tipo de paquete del que se trata. Puede indicar que es un paquete de *Advertising* o de datos.

- **Longitud**: Indica la longitud del campo de datos que viene a continuación.

- **Datos (Payload)**: Contiene la información que se transmite en la comunicación. Pueden ser datos de un dispositivo que está emitiendo *Advertisings* (Broadcaster), pueden ser datos de un Scan Response o los datos que se envían en una conexión entre dispositivos.

- **CRC**: Es el Cyclic Redundancy Check, se trata de un código de detección de errores. En sistemas de hardware binario son fácil de implementar y analizar, y son bastante efectivos para detectar los errores ocasionados por ruido en el canal de transmisión.

2.3 Tipos de Advertisings

Encontramos diferentes tipos de *Advertisings*, cada uno con un objetivo diferente. Éstos sirven para ofrecer contenido como mensaje, información o publicidad hacia otros dispositivos que se encuentren escaneando. También sirven para anunciar permisos de conexión. Cuando entre dos dispositivos se quiere establecer una conexión, uno de ellos ha de darse a conocer mediante este tipo de paquetes, enviando información sobre la conexión y el receptor tiene la opción de aceptar o rechazar el mensaje.

Anteriormente ya hemos comentado por encima que hay diferentes tipos de *Advertisings*. En la tabla 2 encontramos un pequeño resumen de éstos.

Advertising Packet Type	*Connectable*	*Scannable*	*Directed*	*GAP Name*
ADV_IND	Yes	Yes	No	*Connectable Undirected Advertising*
ADV_DIRECT_IND	Yes	No	Yes	*Connectable Directed Advertising*
ADV_NONCONN_IND	No	No	No	*Non-connectable Undirected Advertising*
ADV_SCAN_IND	No	Yes	No	*Scannable Undirected Advertising*

Figura 11: Tipos de Advertisings

A continuación, se describen los 4 tipos de *Advertisings* utilizados en este proyecto junto con sus características y funcionamiento.

2.3.1 Connectable Undirected Advertising

Este tipo es el más común y genérico de todos, ya que éstos no son paquetes dirigidos a un dispositivo BLE concreto y son conectables. Un dispositivo que actúe como Master puede conectarse a otro periférico que se encuentre en estado de *Advertising* enviando este tipo de paquetes, es decir, cualquier dispositivo que escuche estos mensajes será capaz de conectarse.

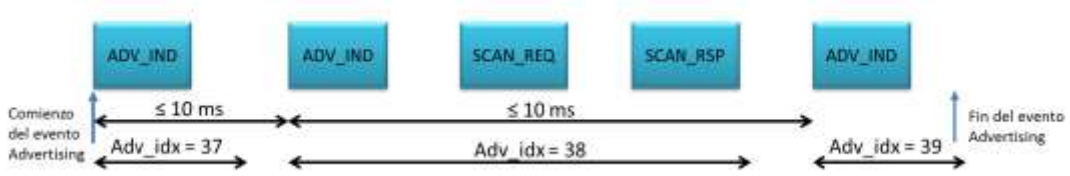

Figura 12: Connectable Undirected Advertising

La figura 12 muestra cómo se repartirían estos paquetes en función del canal. En este caso encontramos un dispositivo escaneando en el canal 38, del cual recibimos una respuesta. El flujo total de mensajes enviados sería el siguiente:

Figura 13: Flujo de mensajes (Connectable Undirected Advertising)

Observamos que la comunicación empieza por el dispositivo que se encuentra enviando los *Advertisings*. El dispositivo que está escaneando detecta su presencia y como quiere establecer una conexión le envía un SCAN_REQ (*Scan Request*) solicitando información adicional al *Advertiser*. Éste le responde con un SCAN_RSP (*Scan Response*) y cuándo el dispositivo que se encontraba escanenado lo recibe, pasa al estado de *Initiating*, prévio a la conexión. Una vez en este estado, el siguiente *Advertising* que recibe se le contesta con un CONNECT_REQ (*Connect Request*), momento en el cual se establece la conexión. El dispositivo que envía esta contestación pasa a ser el Master de la conexión y el otro pasa al rol de Slave.

2.3.2 Connectable Directed Advertising

Este evento se utiliza cuando se quiere establecer una conexión con dispositivo BLE concreto. Cuando un dispositivo recibe este tipo de PDU, puede contestar directamente con un Connect Request para crear la conexión con el Advertiser que se está anunciando.

Se utilizan cuando se quiere conectar con un dispositivo conocido y de forma rápida. En la figura 14 se muestra la distribución de los paquetes según el canal.

Figura 14: Connectable Directed Advertising

Estos paquetes han de contener las direcciones del *Advertiser* y del *Initiator*. La separación temporal entre estos paquetes ha de cumplir una cierta normativa. Dos paquetes de este tipo de *Advertising* enviados en el mismo canal tienen que estar separados unos 3,75ms como máximo. Esto provoca que solamente se permita un escaneo de 3,75ms para detectar los *advertisings*. Este tiempo es desde que se envía el primer *Advertising* en el canal 38, hasta que se vuelve a enviar otro *Advertising* en ese mismo canal.

Figura 15: Flujo de mensajes (Connectable Directed)

Este método provoca que se congestionen más rápidamente los canales, es por eso que sólo se permite utilizar este tipo de *Advertisings* durante un tiempo inferior a 1,28s. Si en éste tiempo no se consigue establecer la conexión, entonces el Controller deja de recibir estos paquetes automáticamente. Una de las principales características de estos eventos es que el dispositivo que escanea ignora cualquier Scan Request que pueda recibir, ya que sus paquetes no tienen datos adicionales, sino que solo contienen las dos direcciones de acceso de los dispositivos implicados en la comunicación.

2.3.3 Nonconnectable Undirected Advertising

Los Nonconnectable Undirected *Advertising* son un tipo de eventos utilizados por dispositivos que no van a establecer una conexión con otro dispositivo. Tampoco saben si están siendo escaneados por otros dispositivos centrales, ni tienen la intención de saberlo. Transmiten estos mensajes de *Advertising* de forma continua y emiten en modo Broadcast (Cualquier receptor puede escuchar sus mensajes). Los dispositivos que se encuentren escaneando en

ese momento sólo pueden escuchar la información emitida por el dispositivo y no pueden conectarse ni solicitar información extra.

Figura 16: Nonconnectable Advertising

El tiempo máximo entre dos paquetes de este tipo de *Advertisings* enviados en el mismo canal es de 10ms. Pueden ser útiles para saber si se ha detectado a una cierta persona que posee un dispositivo BLE o muchas otras utilidades como para corredores de pista cerrada o ciclismo en pista.

2.3.4 Scannable Undirected Advertising

Éste último tipo de *Advertising* es muy parecido al anterior, pero incorpora que es escaneable. Igual que en el anterior, no se puede establecer una conexión pero una vez recibido el *Advertising*, por parte del dispositivo que se encuentra escanenado, se puede enviar un *Scan Request* solicitando más información. El dispositivo que estaba enviando los *advertisings*, lo recibiría y le enviaría esa información solicitada mediante un *Scan Response*.

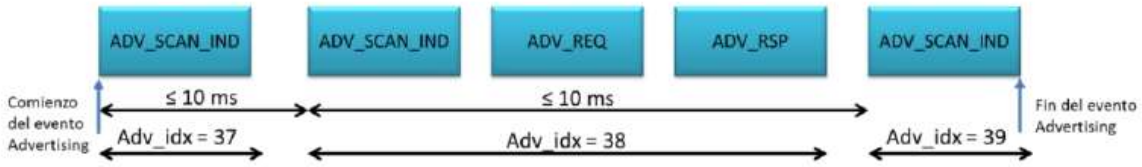

Figura 17: Scannable Undirected Advertising

El tiempo máximo durante el cual se establece el intercambio de información en un canal ha de ser inferior o igual a 10ms, como se observa en la figura 17. Este tipo de *advertisings* son muy útiles, ya que se puede solicitar información extra sin la necesidad de establecer una conexión.

CAPÍTULO 3. DESCRIPCIÓN DEL PROCESO DE SCANNING

El estado de *Scanning* es el encargado de escuchar el medio y captar los paquetes de *Advertising* que están siendo enviados por otro dispositivo. Encontramos dos tipos de *Scanning*, el *Active Scanning* y el *Pasive Scanning*. El escaneo activo escucha los paquetes de *Advertising* y cuando detecta uno, tiene la posibilidad de enviar paquetes *Scan Request* solicitando más información en la comunicación. Esta será recibida mediante un *Scan Response*, mientras que el escaneo pasivo solo es capaz de escuchar paquetes de *Advertising* sin poder solicitar información adicional.

Cuando un dispositivo se encuentra en estado de *Scanning*, solo puede pasar al estado *Standby,* como se ha ejemplificado en la figura 4. Cuando el dispositivo acaba el proceso de escaneo, pasa automáticamente al estado de descanso. La duración en este estado depende de cómo hagamos la configuración del *Scan Interval* y *Scan Window.* El tiempo que nuestro dispositivo se encontrará escaneando es igual a *Scan Window*, por lo tanto, la diferencia que existe entre *Scan Interval* y *Scan Window* cada *Scan interval* será el tiempo en el cual nuestro dispositivo se encontrará en *Standby*.

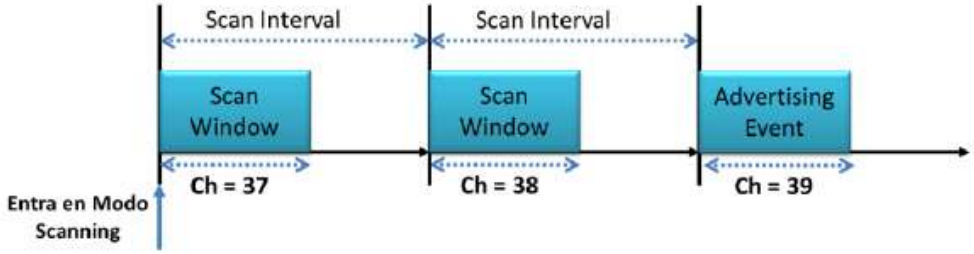

Figura 18: Tiempo de escaneo

Por lo tanto, si conseguimos configurar éstos parámetros de tal manera que el valor de Scan Window sea igual al valor de Scan Interval, conseguiríamos que nuestro dispositivo nunca entrara en estado de descanso y estuviera escaneando continuamente sin pausa.

Esto se ha probado en el laboratorio y al igualar estos dos valores observamos que el comportamiento no es del todo como se ha descrito en el párrafo anterior. Cuando éstos parámetros tienen el mismo valor, observamos que el dispositivo realiza unas pequeñas pausas (entra en estado de Standby) y que estas pausas dependen del modelo del dispositivo que se está utilizando. Por lo tanto, llegamos a la conclusión que depende el fabricante del dispositivo la pausas que realizará serán de mayor o menor duración.

3.1 Passive Scanning

Según las especificaciones, para configurar los parámetros de escaneo, hemos de utilizar el comando ***LE Set Scan Parameters Command*** [4]. Para crear un Scanning personalizado se tienen que configurar los parámetros que se muestran en la tabla TAL que contituyen este comando.

Command	OCF	Command parameters	Return Parameters
HCI_LE_Set_S-can_Parameters	0x000B	LE_Scan_Type, LE_Scan_Interval, LE_Scan_Window, Own_Address_Type, Scanning_Filter_Policy	Status

Figura 19: Parámetros para configurar el Scanning

El primer campo es LE_Scan_Type, se trata de un campo de 1 byte el cual solo tiene dos opciones (0x00 o 0x01). El valor cero indica que el escaneo que se va a realizar es de tipo pasivo y el valor 1 significa que el escaneo será activo.

Los dos siguientes campos LE_Scan_Interval y LE_Scan_Window son campos de dos bytes de longitud. Estos campos están limitados entre 2.5 ms y 10.24 s, en pasos de 0.625 ms, que posteriormente acabamos convirtiendo a hexadecimal. Si se quisiera configurar este parámetro en 100ms, se tendría que convertir la división 100/0.625 ms = 160, por lo tanto este resultado en hexadecimal sería 0x00A0. Este sería el número a configurar en el parámetro deseado. Una de las condiciones que imponen las especificaciones es que el Scan Interval siempre ha de ser superior o igual a Scan Window. En el caso que fuera igual, tendríamos a nuestro dispositivo escaneando continuamente, con la condición que hemos comentado anteriormente.

Own_Address_Type tiene una longitud de un byte, este parámetro indica el tipo de dirección que va a utilizar el dispositivo en los paquetes de Scan Request, esta puede ser pública o aleatoria.

El último campo, Scanning_Filter_Policy se utiliza para establecer una política en el filtrado de paquetes que se reciben. Si se configura con cero, se aceptarán todos los paquetes recibidos, sin ningún filtrado. Sin embargo, utilizando la lista podemos aceptar solo los paquetes que vienen de un dispositivo en concreto o contrariamente podemos rechazar los paquetes de *Advertising* que vengan de un dispositivo.

Por lo tanto, si se quisiera realizar la configuración del Pasive Scanning de forma contínua, con un valor de 100 ms tanto para Scan Window como para Scan Interval, con una dirección de tipo pública y con una política de filtro que acepte todos los paquetes sería la que encontramos en el cuadro de más abajo. Para configurar los dispositivos y modificar los parámetros que los componen se han utilizado los comandos HCI [5].

```
sudo hcitool –i hci0 cmd 0x08 0x000B 00 A0 00 A0 00 00 00
```

Teniendo en cuenta que los campos están en el orden que se han explicado. Se pueden dar muchos más casos de diferentes configuraciones, haciendo una simple variación en cualquier parámetro.

3.2 Active Scanning

Para conseguir un Active Scanning[1], hay que seguir los mismos pasos explicados en el apartado 3.1. Los parámetros para la configuración son exactamente los mismos, con la única diferencia que el parámetro LE_Scan_Type del comando HCI_LE_Set_Scan_Parameters tiene que ser 0x01.

Por lo tanto, la secuencia de comandos que hay que seguir para la configuración del Active Scanning es la siguiente:

```
sudo hcitool –i hci0 cmd 0x08 0x000C 00 00
sudo hcitool –i hci0 cmd 0x08 0x000B 01 A0 00 A0 00 00 00
sudo hcitool –i hci0 cmd 0x08 0x000C 01 00
```

[1] Ver Anexo 1 de Scripts, para saber cómo configurar el Scanning

CAPÍTULO 4: PRUEBAS CON DISPOSITIVOS BLE

El dispositivo utilizado se trata del Red Bear Nano, un dispositivo BLE con un consumo muy bajo y el más pequeño que pertenece a esta versión 4.1 de Bluetooth. Con este dispositivo es muy fácil producir prototipos para el IoT u otros proyectos interesantes. Nano puede funcionar entre 1.8V y 3.3V, compatible con muchos componentes electrónicos.

Figura 20: Dispositivo Red Bear BLE Nano

Para llegar a saber cuál es la configuración deseada de nuestro dispositivo, se han realizado diferentes experimentos. Primero de todo se ha hecho una caracterización del dispositivo, para saber cuál es su diagrama de radiación, parámetro importante para saber dónde colocar nuestro dispositivo en las cursas, teniendo en cuenta la máxima distancia por la cual puede pasar un corredor. También se ha hecho un test de para caracterizar el consumo de energía del dispositivo BLE Red Bear Nano [6] con diferentes configuraciones.

4.1 Caracterización Dispositivo BLE

Este experimento se ha realizado con la finalidad de ver como el dispositivo BLE Red Bear Nano radia en todas sus direcciones. Para ello se ha situado el transmisor en el centro (dispositivo BLE) y separados una distancia de 2 metros en radio, se han ido haciendo capturas desde 12 puntos diferentes como se muestra en la figura 21. El experimento se ha realizado en una zona abierta y lejos de cualquier otro dispositivo que pueda trabajar en la misma frecuencia provocando así interferencias o pérdidas de paquetes. La posición 1 se corresponde con 0º y conforme vamos aumentando una unidad, el ángulo girado es de 30º.

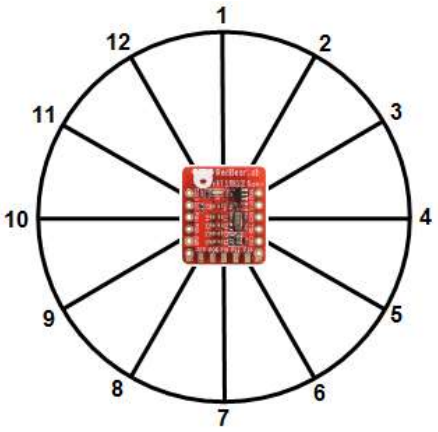

Figura 21: Puntos para la caracterización del dispositivo

El experimento se ha realizado dos veces, situando el dispositivo en dos posiciones diferentes, vertical y horizontal, para ver si de ambos modos la potencia recibida en cada punto es similar. En cada ángulo se han analizado las capturas obtenidas y se ha procedido al cálculo de la potencia media.

4.1.1 Caracterización en Vertical

En la figura 22 observamos el diagrama de radiación del dispositivo BLE cuando lo situamos en una posición vertical. .

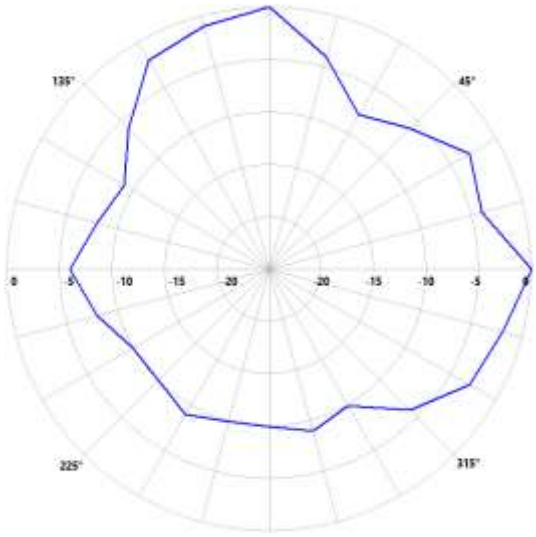

Figura 22: Diagrama radiación BLE Red Bear Nano (pos. Vertical)

La máxima diferencia entre el “mejor” ángulo y el “peor” no es superior a 6 dB. Podemos decir que el diagrama de radiación del dispositivo es prácticamente omnidireccional y que capta por todos sus ángulos la misma cantidad de paquetes.

4.1.2 Caracterización en horizontal

Para el caso del dispositivo en horizontal los resultados obtenidos comparten muchas similitudes. Observamos también que la máxima diferencia que existe entre el valor máximo y mínimo es de inferior 6 dB. Valor que no influirá a la hora de captar nuestros paquetes, ya que nuestro dispositivo estará situado cerca de los corredores y la potencia que nos llega será suficiente.

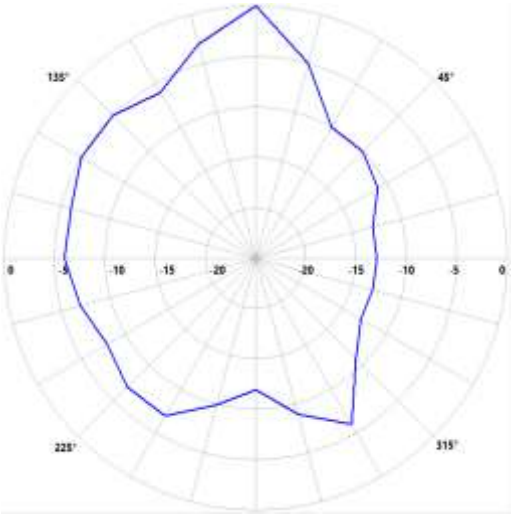

Figura 23: Diagrama radiación BLE Red Bear Nano (pos. Horizontal)

Por lo tanto, después de observar la potencia captada por el dispositivo en dos posiciones diferentes y que en ambos casos la potencia recibida sea prácticamente la misma, llegamos a la conclusión que la posición del dispositivo es prácticamente indiferente para nuestro objetivo de detectar a corredores cuando pasan por la zona de cobertura, ya que su diagrama de radiación es prácticamente omnidireccional.

4.2 Caracterización de Energía

El objetivo de este test es caracterizar el consumo de energía de nuestro dispositivo con diferentes configuraciones. Para ello se han llevado a cabo dos pruebas, la primera en la cual se ha analizado el consumo para una configuración de tiempo entre paquetes de 100ms, dónde se ha ido variando el tamaño del paquete entre 26 Bytes, 10 Bytes y 1 Byte y también para diferentes potencias.

La segunda parte del experimento se centra solamente en los paquetes separados 300 ms y 500 ms, variando también el tamaño del paquete pero manteniendo la potencia fija a 4 dBm.

4.2.1 Consumo configuración a 100ms

La figura 24 muestra el consumo del dispositivo para una configuración de un paquete de tamaño de 26 Bytes, variando la potencia entre -40 dBm y 4 dBm. Obtenemos que el consumo medio para un paquete de 26 Bytes es 21,46µA.

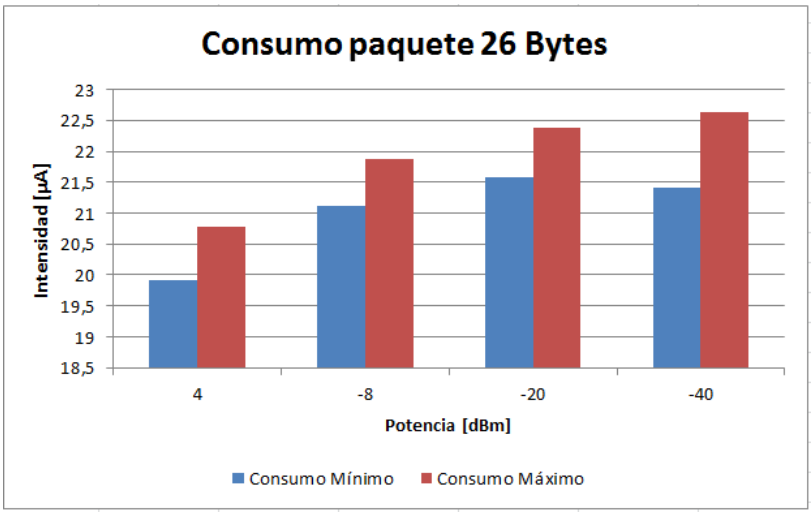

Figura 24: Consumo paquete 26 Bytes (100ms)

La figura 25 muestra el consumo del dispositivo para un tamaño de paquete de 10 Bytes, variando también la potencia entre los mismo valores. El consumo medio que se obtiene es de 20.075 µA.

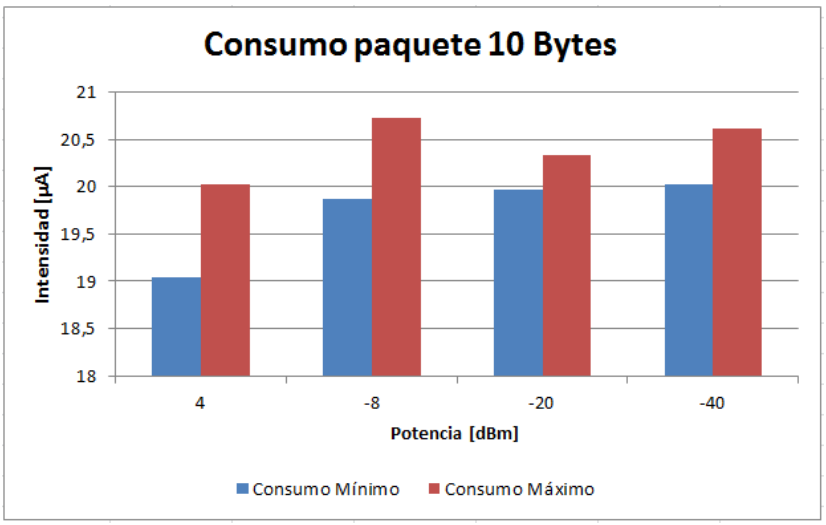

Figura 25: Consumo paquete 10 Bytes (100ms)

Finalmente, la figura 26 muestra el consumo par aun tamaño de paquete de 1 Byte cada 100ms. El consumo medio para esta configuración es de 18.82 µA.

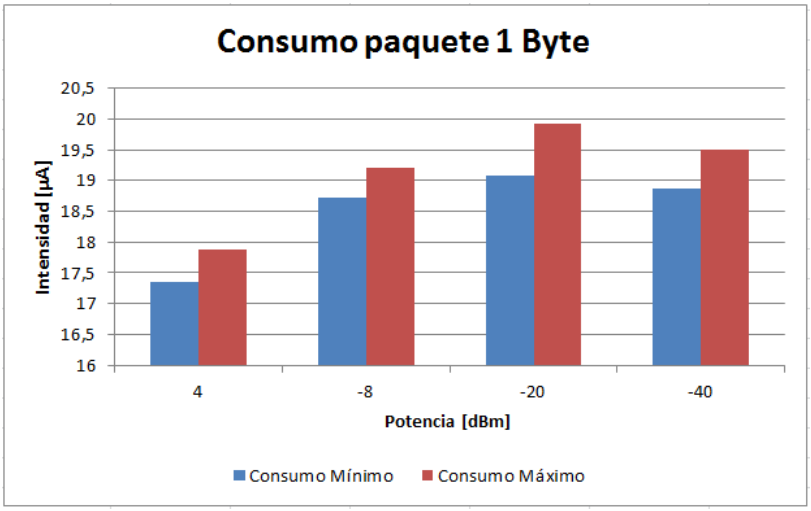

Figura 26: Consumo paquete 1 Byte (100ms)

Se observa que conforme bajamos el tamaño del paquete hasta 1 Byte, el consumo medio del dispositivo disminuye prácticamente 3 µA. Esto se debe a que el dispositivo no necesita consumir tanta energía para procesar un paquete de menor tamaño. Un dato que da para pensar es que contra menor potencia tenga programada el dispositivo, mayor consumo tiene este.

4.2.2 Consumo configuración 300ms y 500ms

Si aumentamos el tiempo entre paquetes consecutivos hasta 300 ms o 500 ms observamos que el consumo del dispositivo se reduce aún más conforme aumentamos la separación entre estos. Esto se debe a que el dispositivo está menos rato procesando los paquetes y puede pasar al estado de Standby. En las siguientes figuras se muestra el consumo para dichas configuraciones.

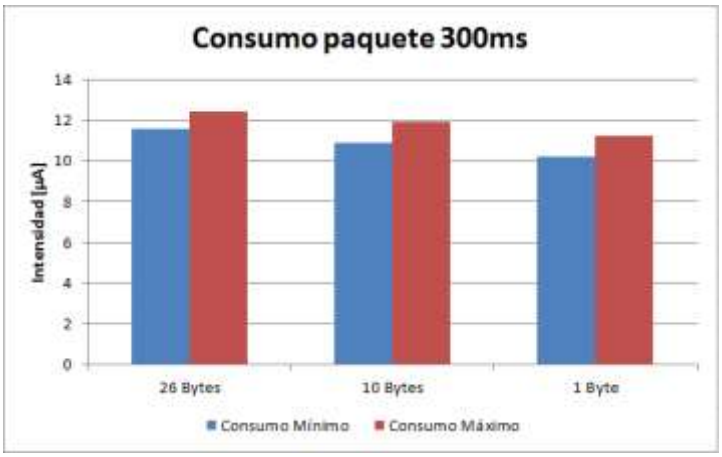

Figura 27: Consumo diferentes tamaños de paquete a 300ms

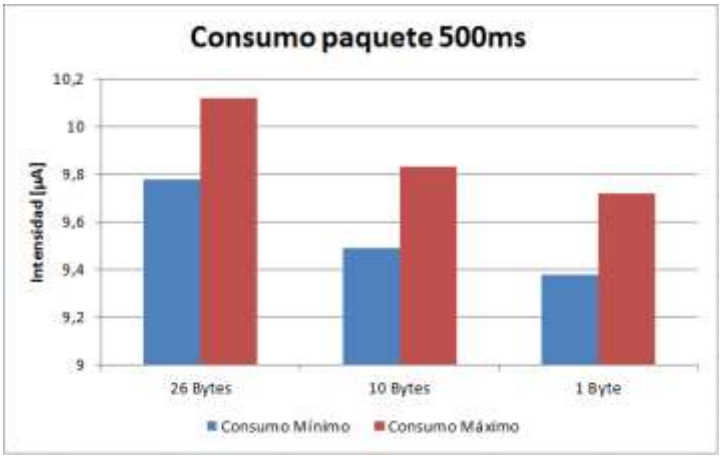

Figura 28: Consumo diferentes tamaños de paquete a 500ms

4.3 Test Cobertura

En este test se calcula la potencia que recibe el dispositivo en función de la distancia. Se situó un dispositivo quieto en una zona haciendo la función de transmisor y con un receptor, en este caso el teléfono móvil, se fueron tomando diferentes capturas a varias distancias del transmisor.

La separación entre cada punto dónde se han obtenido capturas es de 1 farola, la cual se puede calcular fácilmente mediante google maps o mediante un calculador de distancias GPS, introduciéndole las coordenadas que obtenemos en la captura desde el teléfono móvil, ya que este estaba preparado para ello.

En este test se puede observar la evolución de la potencia en función de la distancia y los paquetes recibidos en cada punto que se ha hecho una captura. En la tabla 4 se puede ver la distancia a la cual se han tomado las medidas, la potencia media que se ha recibido y el número de paquetes.

	Inicial	1	2	3	4	5	6	7	8	9	10
Distancia [m]	0	5,235	20,26	34,48	49,21	63,66	78,04	92,46	107,17	121,42	136,04
Pot Media	-59,278	-75,168	-81,642	-94,602	-94,804	-89,968	-90,319	-92,807	-94,925	-95,465	-97,318
Nºpaq	90	143	109	93	56	154	69	83	120	99	66
Duración (s)	52	95	60	77	40	45	50	56	33	89	85
paq/s	1,731	1,505	1,817	1,208	1,400	3,422	1,380	1,482	3,636	1,112	0,776

Figura 29: Datos Test Cobertura

En la gráfica de la figura 30 observamos que cada nube de datos se corresponde a cada punto dónde se ha tomado la medida y la potencia con la que se ha recibido cada paquete a esa distancia.

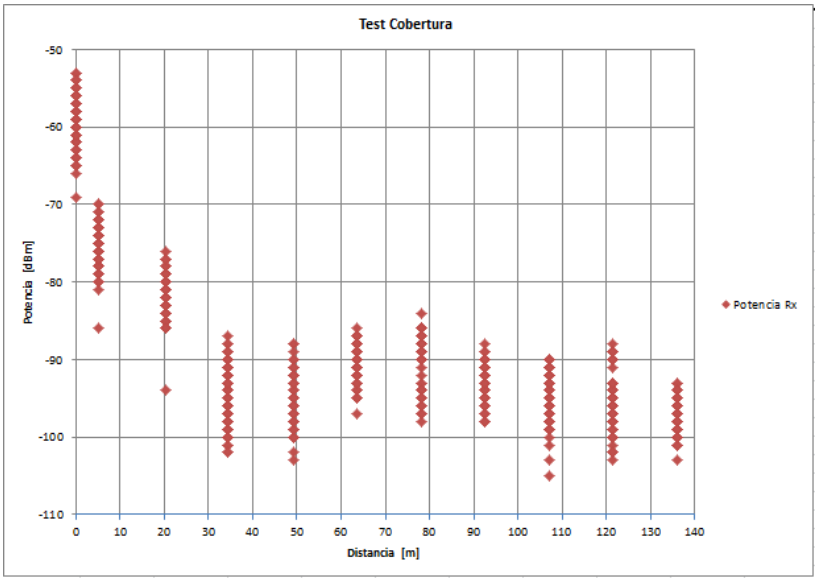

Figura 30: Evolución de la potencia en función de la distancia

La Figura 31 se corresponde a la evolución de la potencia en función de la distancia, para ello se ha hecho el cálculo de la potencia media que se recibe en cada punto, teniendo en cuenta el total de paquetes recibidos.

Observamos que la potencia va disminuyendo conforme nos vamos alejando del transmisor, pero observamos que a unos 60 metros recibimos más potencia que a 35 metros por ejemplo, esto se debe al modelo de tierra plana, ya que a 35 metros del transmisor podemos estar en un punto dónde las ondas se cancelan entre sí, interferencia destructiva. Mientras que a 60 metros podemos haber recibido a parte del rayo directo la suma de los rebotes que contribuyen a la recepción de una mayor potencia.

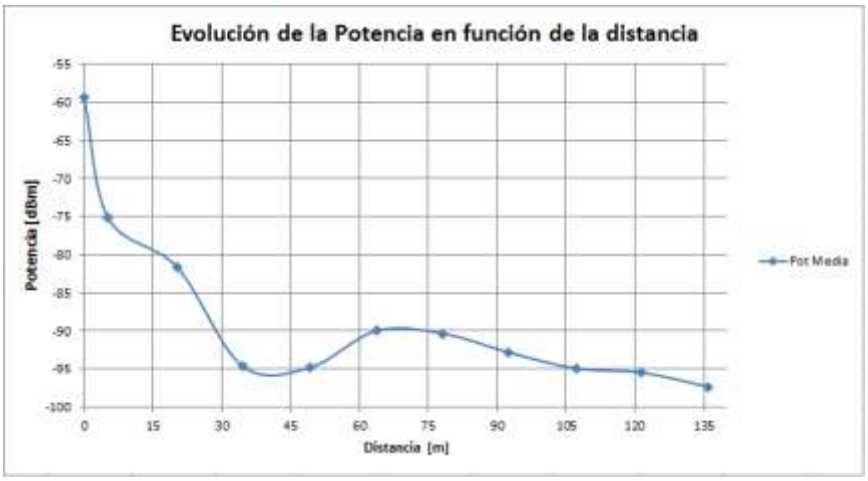

Figura 31: Evolución de la potencia en función de la distancia

4.4 Test Advertising Scannable

La finalidad de este test es ver el comportamiento de este tipo de *Advertising* y todos los posibles casos a los cuales nos podemos enfrentar, es decir, los inconvenientes que nos encontraremos al realizar una comunicación entre dos o varios dispositivos, problemas de propagación, configuración, entre otros.

Este test fue realizado con dos transmisores, cada uno conectado a un señor de corriente y un receptor. Los transmisores se tratan de dispositivos SENA Parani-UD100 (ver [7]), que también incorporan la versión 4.0 de bajo consumo. Sin embargo, los dos dispositivos receptores para los que se han realizado las pruebas son diferentes, Belkin y Trust, ya que el comportamiento de ambos dispositivos difiere de uno al otro. Las capturas realizadas mediante el osciloscopio del laboratorio son con un sensor para cada dispositivo y un canal para cada uno también. Esto es para poder diferenciar mejor el consumo de cada dispositivo en cada instante de tiempo, la cual cosa facilita mucho el entendimiento de cada proceso.

4.4.1 Dispositivo BELKIN

El dispositivo Belkin se trata de un adaptador USB Bluetooth, cuya versión es la 4.0 Bluetooth. El modelo del dispositivo utilizado es el F8T065BF. Después de hacer un análisis exhaustivo de todos los posibles casos durante semanas, se han seleccionado varios casos en los que se explicará que está pasando en esa comunicación.

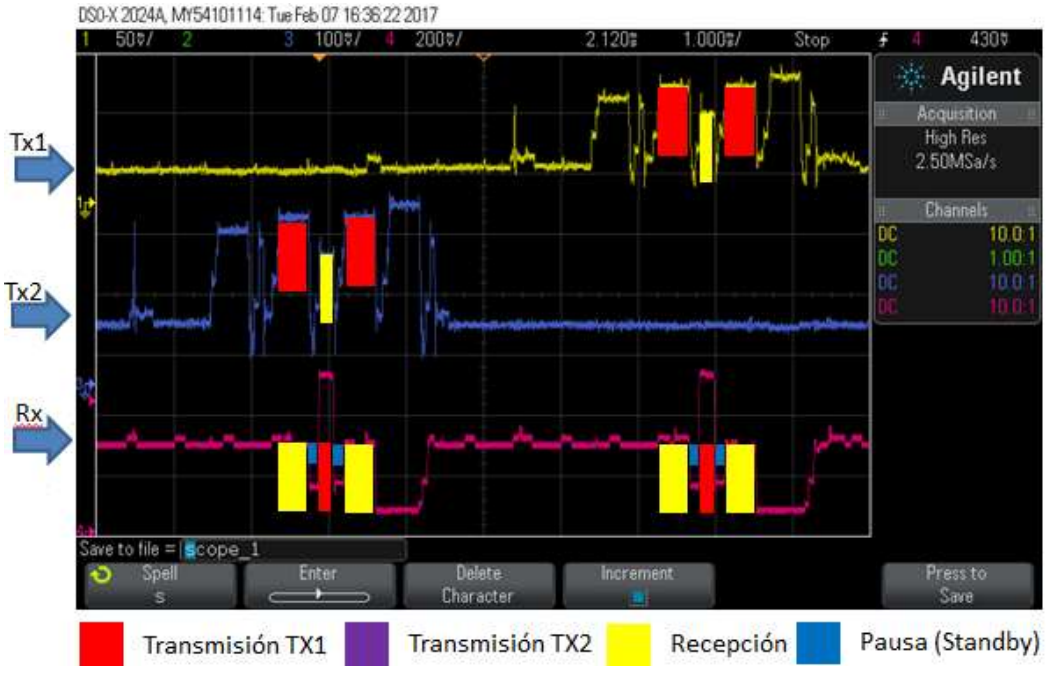

Figura 32: Comunicación Belkin ADV_DISCOVERABLE Canal 38

En este ejemplo de la figura 32, observamos que el receptor está captando los paquetes del canal 38. Primero le llega un paquete de transmisor 2 (azul), que

se procesa correctamente, se envía en SCAN_REQ y se le responde con el SCAN_RSP, siguiendo el proceso normal. Más adelante le llega otro paquete del transmisor 1 (amarillo) que también se acaba procesando. Este sería el caso ideal, en el cual no existen problemas de colisión, ya que los paquetes llegan separados un cierto tiempo y se pueden recibir ambos correctamente.

No siempre los paquetes llegarán separados el tiempo suficiente para que el receptor pueda demodular los dos. Por lo tanto, también se pueden dar otros casos que se explican en las siguientes figuras.

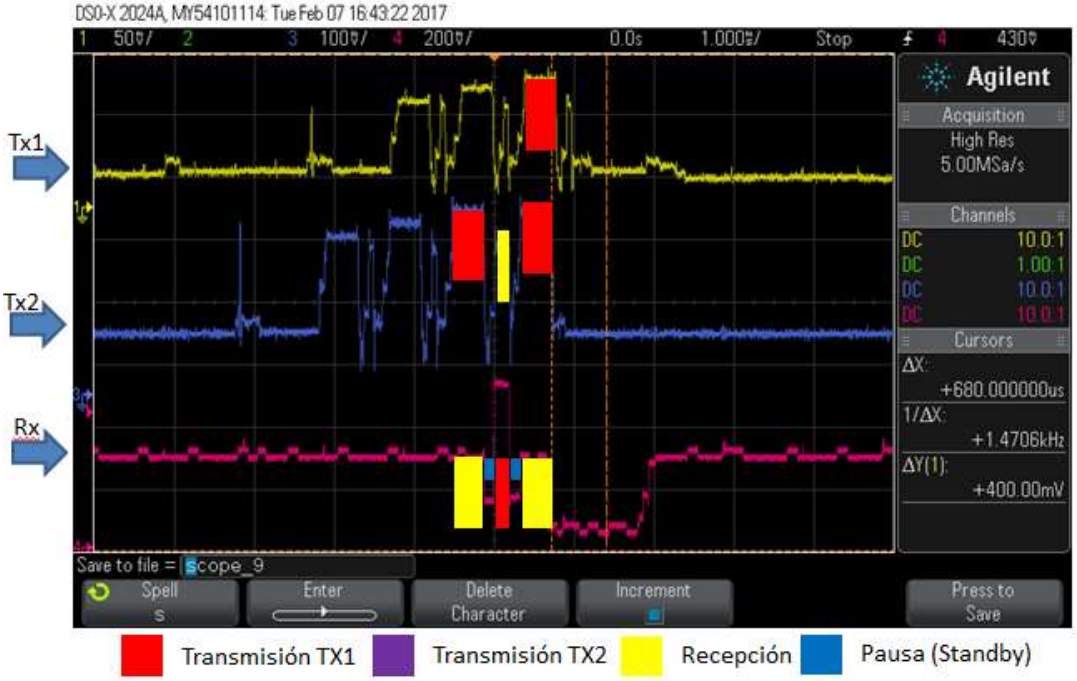

Figura 33: Comunicación Belkin ADV_DISCOVERABLE (Efecto Captura)

Lo que ocurre en la figura 33 es que el dispositivo receptor está escaneando en la frecuencia del canal 39, le llega primero el paquete de transmisor 2 (azul) y lo empieza a procesar, se espera un tiempo y se envía el SCAN_REQ. El transmisor 2 lo recibe y le manda un SCAN_RSP que también es bien recibido y procesado. El transmisor 1 emite su paquete del canal 39 cuando el receptor está recibiendo el SCAN_RSP de transmisor 2, aun así el receptor acaba de procesar el paquete (Efecto captura) y del transmisor 1 no recibe nada. Esto se debe a que el paquete que ha recibido primero tenía una potencia superior al de transmisor 1 y por lo tanto lo acaba procesando.

4.4.2 Dispositivo TRUST

El dispositivo Trust utilizado también es un adaptador USB Bluetooth low energy. El modelo es el Trust 18187 (Ver características [8]), para cualquier duda se pueden consultar las especificaciones del dispositivo.

Con este dispositivo se realizaron las mismas pruebas, pero se han tenido en cuenta una problemática diferente a la del otro dispositivo, dando a conocer casos diferentes de los anteriores. En la figura 34 se observa una comunicación con el dispositivo Trust dónde se puede apreciar que el paquete del segundo transmisor no acaba de ser bien recibido.

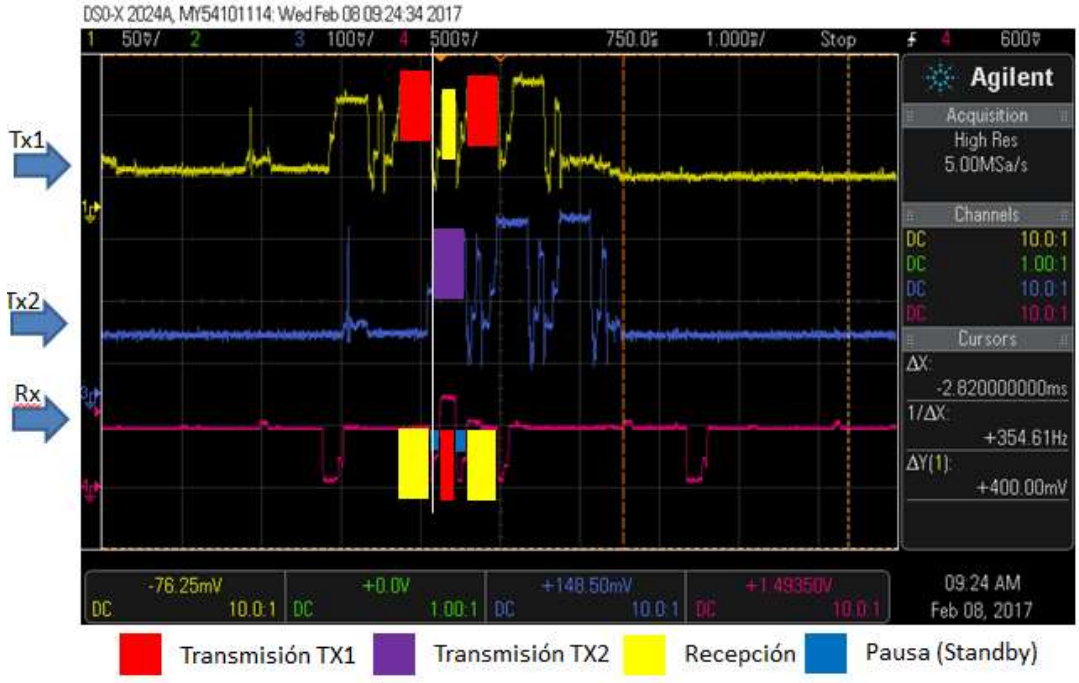

Figura 34: Comunicación Trust pérdida paquete

En la figura anterior se observa cómo se recibe el paquete del canal 38 del transmisor 1, este se procesa y se espera un tiempo. Posteriormente se envía un SCAN_REQ hacía el transmisor del paquete. Este lo recibe y responde con un SCAN_RSP, que el receptor lo procesa correctamente. El receptor añade un gap posterior al procesado, que podemos observar que es de duración diferente a la del dispositivo anterior (Belkin). El segundo transmisor emite su paquete, pero no es recibido por el dispositivo que se encontraba escaneando, ya que cuando ha mandado el paquete el receptor estaba en una pausa.

Este es un ejemplo dónde no se acaba de procesar el paquete, aunque los dos transmisores han mandado el suyo. La “culpa” es del receptor que no es capaz de procesar ambos paquetes.

La siguiente figura muestra cómo se interrumpe el proceso de comunicación por la llegada de otro paquete justo cuando se comienza a recibir el SCAN_RSP. La colisión se produce en el canal 37, que es la primera muestra de consumo en la señal del transmisor 1.

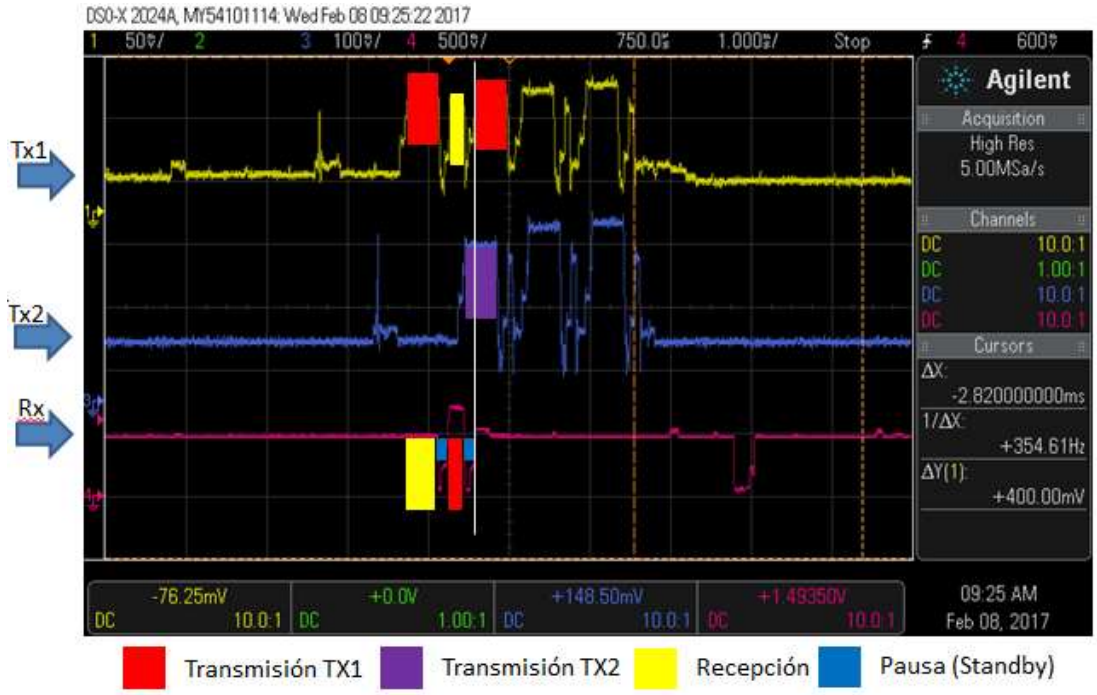

Figura 35: Comunicación Trust colisión canal 37

En la figura 35 se observa cómo el receptor comienza a recibir el paquete del canal 37 del transmisor 1. El receptor solicita información extra mediante un SCAN_REQ al transmisor del paquete. Este recibe su mensaje y le responde con un SCAN_RSP, pero cuando el receptor pretende recibir su mensaje, el transmisor 2 empieza a enviar su paquete de *Advertising* del canal 37 provocando así una colisión. El receptor no es capaz de diferenciar entre los paquetes y vuelve al estado de *scanning*.

Por lo tanto, cómo el receptor no recibe el SCAN_RSP que le pretende enviar el transmisor 1, justo cuando el transmisor 2 envía su *Advertising*, no se acaba demodulando ningún paquete.

Una vez analizados ambos dispositivos, en diferentes situaciones, observamos que los gaps que introduce un dispositivo son de duración y en posiciones diferentes a los del otro dispositivo [9]. Esto se debe a los componentes de cada dispositivo, es decir, la duración de los gaps que tenga cada dispositivo dependen del fabricante.

Lo que comparten estos dispositivos, a parte de la versión 4.0 de bajo consumo es que siempre introducen un gap después del procesado de cualquier paquete de Advertising.

4.5 Test con dispositivos en movimiento

Este test tiene la finalidad de observar el comportamiento de varios dispositivos BLE cuando uno de ellos está en movimiento. Para observar la capacidad de detección de éstos, se han realizado los siguientes experimentos. Cuatro dispositivos transmitiendo dentro de un coche que circula a una velocidad aproximada de 120 km/h y un receptor situado en un punto superior, a una altura de 5 metros, de los dispositivos, con la intención de observar cuando varia la potencia debido al movimiento. Y otro test con 52 dispositivos, mediante el cual se pretende observar la capacidad de detección de nuestro dispositivo Red Bear nano.

4.5.1 Test 4 Dispositivos

Para este test se configuraron los dispositivos con paquetes de un tamaño de 26 Bytes, cada 100ms y con una potencia de 4 dBm.

En la figura 36 observamos la RSSI captada por el receptor durante la primera pasada. Se puede apreciar que conforme nos vamos acercando al dispositivo vamos captando mayor potencia y conforme se aleja ésta va disminuyendo. La RSSI recibida para el resto de pasadas es muy similar a la de esta figura.

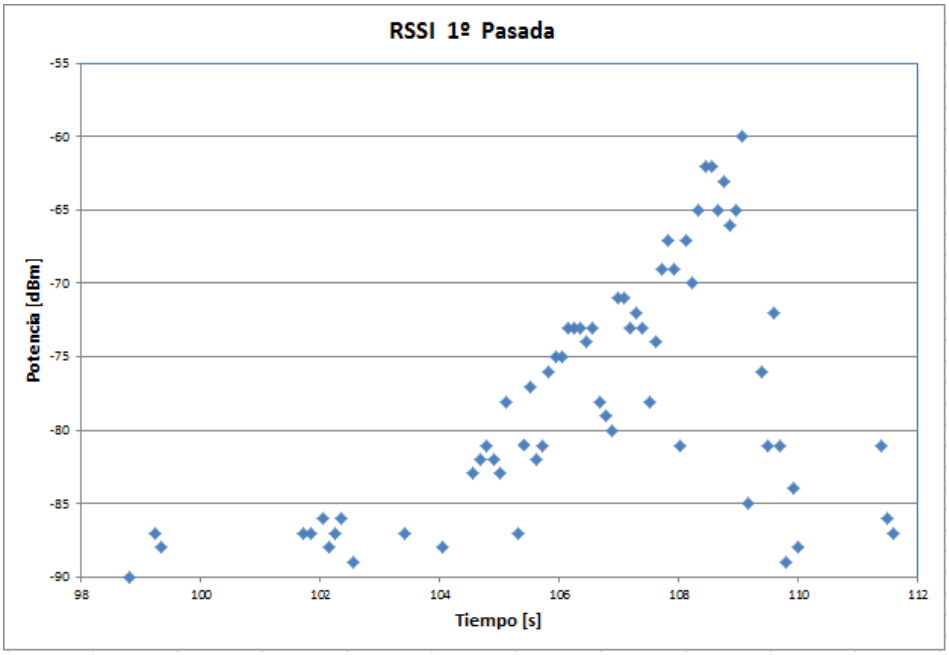

Figura 36: RSSI Primera pasada en movimiento

Esta gráfica muestra la evolución de la potencia en función del tiempo. El primer punto que observamos es cuando el receptor detecta por primera vez a uno de los cuatro receptores y poco a poco va detectando al resto varias veces y con mayor potencia. Sabiendo la velocidad y el tiempo que ha tardado el dispositivo en detectarlo, podemos saber que el primer dispositivo lo ha detectado a una distancia de 300 metros aproximadamente.

La siguiente gráfica se corresponde a la captura de todos los paquetes en las 4 pasadas que hicimos. Se puede observar las cuatro nubes de puntos captadas en tiempos muy próximos y con potencias muy diferentes debido al movimiento, provocado por el efecto doppler
.

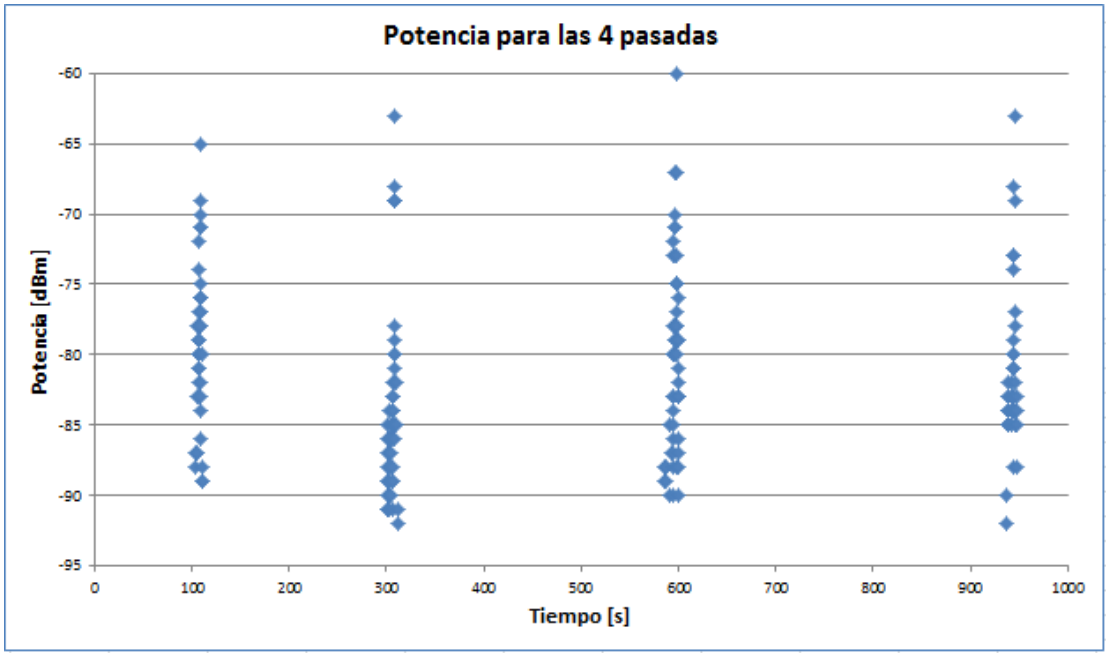

Figura 37: Potencia recibida para cada pasada

La separación en tiempo de cada pasada es indiferente para la conclusión a la cual se quiere llegar con este experimento, ya que el tiempo de ida y vuelta puede depender de otros factores externos al experimento.

La finalidad de este test es ver cuánto varía la potencia captada si uno o varios de los dispositivos se encuentran en movimiento. En los resultados obtenidos se observa que puede llegar a variar mucho si la velocidad a la cual va el dispositivo es muy elevada, ya que se pueden perder paquetes que en un sistema estático no ocurriría. En nuestro caso, un corredor de maratón puede llevar una velocidad de 30 km/h (dependiendo del corredor), por lo tanto, la variación de potencia que tendríamos sería mucho menor a la obtenida en esta prueba.

4.5.2 Test 52 Dispositivos

Este test con 52 dispositivos tiene la finalidad de ver si nuestro dispositivo es capaz de capturar todos los dispositivos cuando nos acercamos a la zona de cobertura corriendo a una velocidad de aproximadamente 25 km/h. Se ha realizado para 4 configuraciones diferentes y ver cuál es la que mejor se adapta a las características buscadas para la detección de corredores. También se introduce un test del cálculo de la distancia máxima que se puede conseguir con cada configuración, para luego comparar como varía esta distancia.

Para llevar a cabo esta prueba, se han programado los 52 dispositivos BLE (BLE Red Bear Nano) con la misma configuración. Para poner todos los dispositivos a la vez en marcha, se han soldado a una placa y ésta se ha alimentado mediante una batería. Una vez puesta en marcha, se espera un tiempo para conseguir que los mensajes de *Advertising* que envía cada dispositivo se aleatorizen. Las 4 configuraciones son las siguientes:

- 1ª Configuración: 26 Bytes, 100ms y 4dBm.
- 2ª Configuración: 26 Bytes, 100ms y -8dBm.
- 3ª Configuración: 26 Bytes, 100ms y -20dBm.
- 4ª Configuración: 26 Bytes, 100ms y -40dBm.

1ª Configuración (26 Bytes, 100ms y 4dBm)

Para ver el alcance máximo de estos dispositivos nos situamos en la calle Esteve Terradas y fuimos alejando los dispositivos hasta una distancia total de 170m y la señal de todos los dispositivos era captada. Como está configurado con una potencia muy alta, es normal que el alcance de la conexión supere los 100 metros. En la figura 38 se observa dónde se situaron los dispositivos para hacer las pruebas.

Figura 38: Primera distancia Test Cobertura 1ª configuración

Para ver el alcance máximo de esta configuración decidimos encontrar un espacio abierto y seguimos los mismos pasos que en el caso anterior, alejar los dispositivos hasta la distancia máxima a la cual se recibían todos los paquetes de los dispositivos. En la figura 39 se puede ver que se consiguió una comunicación de hasta 400 metros de distancia, detectando a todos los dispositivos.

Figura 39: Distancia máxima Test Cobertura 1ª configuración

Una vez realizada esta pequeña prueba, pasamos al test en movimiento con los 52 dispositivos y ver cuánto varía el radio de cobertura. Para ello, se situaron los 52 dispositivos en un punto estático y era el receptor el que se movía, factor que no altera las medidas, ya que mover los 52 dispositivos es más incómodo. Para la velocidad de 25km/h del receptor, obtenemos que nuestro radio de cobertura ha sido reducido notablemente hasta una distancia de 186 metros y aun así sigue siendo una zona de cobertura muy grande.

Figura 40: Distancia máxima en movimiento 1ª Configuración

2ª Configuración: (26 Bytes, 100ms y -8dBm)

Para esta segunda configuración se han realizado las mismas pruebas que en la anterior, exceptuando que para la prueba en movimiento no ha sido necesario un espacio tan abierto, ya que al reducir la potencia considerablemente se ha reducido también el alcance.

Los resultados obtenidos son que el radio de cobertura aproximado para los dispositivos en estático es de aproximadamente 160m. Esta distancia sigue siendo demasiado grande comparado con la máxima distancia a la cual se podría encontrar un corredor en cualquier cursa o maratón de las zonas de control, ya que los recorridos por dónde se realizan estos eventos no suele haber tanta distancia entre transmisor y receptor.

La prueba en movimiento es exactamente la misma, obteniendo que el receptor sea capaz de detectar a todos los dispositivos cuando se pasa a una distancia de 10 metros en perpendicular al eje de los 52 dispositivos. Por lo tanto, la potencia utilizada con esta configuración sigue siendo un poco elevada, pero no se descarta esta configuración como útil.

Figura 41: Distribución dispositivos Test movimiento

3ª Configuración: (26 Bytes, 100ms y -20dBm)

Con esta configuración se ha obtenido una cobertura máxima de aproximadamente 23m, distancia a la cual se detectaban prácticamente los 52 dispositivos. Este radio de cobertura se ajusta mucho más al objetivo buscado con esta prueba, ya que para una distancia de 20 metros seríamos capaces de detectar a los 52 dispositivos o más.

Al realizar el mismo test pero en movimiento, obtenemos que al realizar una pasada a la velocidad de 25km/h y a una distancia perpendicular al eje del dispositivo de 10m, todos los dispositivos son detectados

Teniendo en cuenta el tamaño de la calle más grande por la cual pasa una maratón, esta es la configuración que más se adapta para nuestro proyecto. Aun así se tendría que acabar de ajustar la potencia para conseguir el radio de cobertura deseado entre esta y la anterior configuración. También se han calculado el máximo número de personas y el máximo número de personas que entran por segundo en nuestra zona de cobertura, que son datos necesarios y que se explican en el análisis de la maratón de Barcelona.

4ª Configuración: (26 Bytes, 100ms y -40dBm)

Teniendo en cuenta que la configuración anterior con una potencia mayor ya necesitaba un poco más de potencia, esta configuración no es la idónea para nuestro objetivo. Aun así se hizo esta prueba, ya que puede tener utilidad para otros deportes o aplicaciones.

El máximo radio de cobertura que se consiguió fue de 6 metros, distancia a la cual se detectan los 52 dispositivos. La prueba en movimiento se ha realizado pasando muy cerca de los dispositivos y a la misma velocidad que para las configuraciones anteriores. El resultado fue que todos los dispositivos fueron detectados, de tal manera que esta configuración podría ser útil para corredores de pista de larga distancia, ya que siempre hacen el mismo recorrido y vuelven a pasar por meta, cerca del dispositivo que se encargaría de detectarlos y obtener sus tiempos.

4.6 Test Discovery Time

El objetivo de este test es calcular el tiempo de descubrimiento necesario para detectar una gran cantidad de dispositivos BLE que están emitiendo mensajes de *Advertising* a la vez. Para ello, se han programado 26 dispositivos todos con la misma configuración y para dos tipos de mensajes de *Advertising*.

Para la primera prueba se han programado todos los dispositivos con el tipo Nonconnectable Undirected *Advertising* (ADV_NONCONN_IND), recordamos que este tipo de *Advertising* no está dirigido a ningún dispositivo ni se puede establecer conexión. Por lo tanto, es dispositivo receptor encargado de detectar a todos los dispositivos solo tiene que recibir un mensaje de *Advertising* de los dispositivos que se están dando a conocer.

Para la segunda prueba se han programado todos los dispositivos con el tipo Scannable Undirected *Advertising* (ADV_SCAN_IND). Este tipo de mensajes de *Advertising* pueden solicitar información extra, la cual cosa retrasa el proceso de detección de los dispositivos.

4.6.1 Discovery Time Nonconnectable Undirected Advertising

En la figura 42 observamos el tiempo de descubrimiento de los 26 dispositivos, calculado 150 veces, con el objetivo de obtener un valor medio de este tiempo.

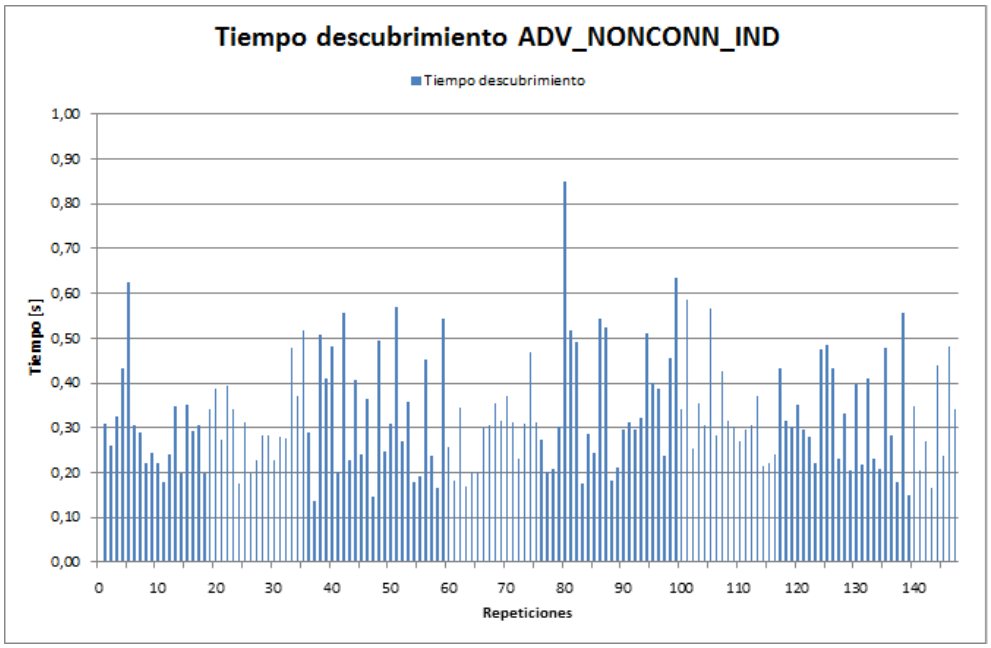

Figura 42: Discovery Time ADV_NONCONN_IND

En la figura 43 observamos el tiempo medio de descubrimiento para este tipo de mensajes de *Advertising*, que se corresponde con **0.3067**s.

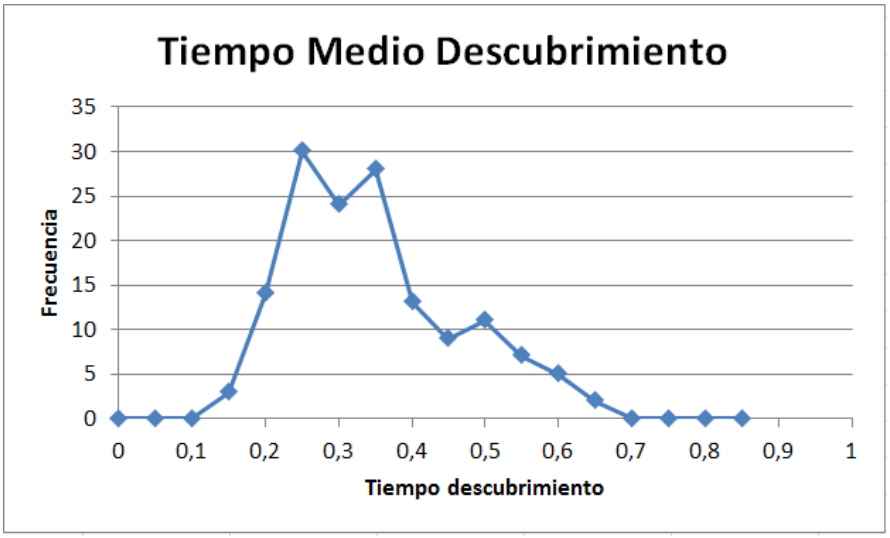

Figura 43: Tiempo medio descubrimiento ADV_NONCONN_IND

4.6.2 Discovery Time Scannable Undirected Advertising

En la figura 44 observamos el tiempo de descubrimiento de los 26 dispositivos, calculado 100 veces, con el objetivo de obtener un valor medio de este tiempo.

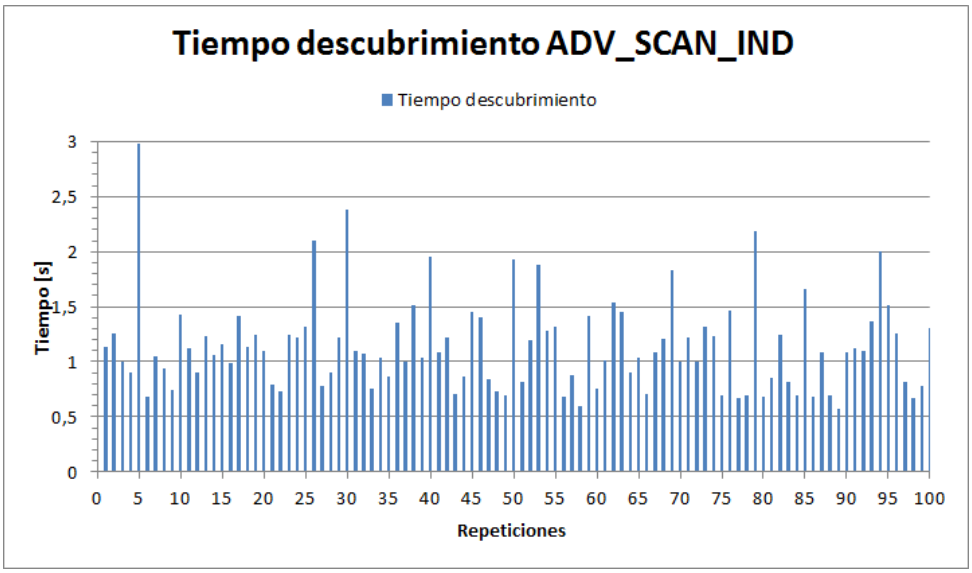

Figura 44: Discovery Time ADV_SCAN_IND

En la figura 45 observamos el tiempo medio de descubrimiento para este tipo de mensajes de *Advertising*, que se corresponde con **1.1381**s.

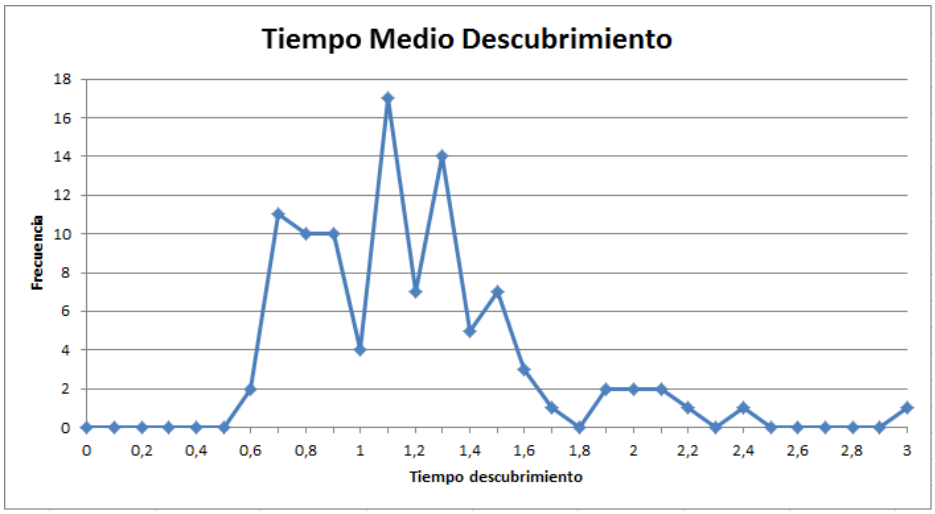

Figura 45: Tiempo medio de descubrimiento ADVSCAN_IND

4.7 Test Potencia en función del Canal

La finalidad de este test es observar el comportamiento de los paquetes que envía cada dispositivo en función del canal. Para ello se ha realizado el test en dos entornos diferentes, uno con ausencia de gente y otro en una sala en la cual la gente se mueve. El primer caso pretende simular una cámara anecoica, que se trata de una sala que está diseñada para absorber prácticamente en su totalidad las reflexiones producidas por los rebotes de las ondas electromagnéticas en cualquier superficie. Para el segundo caso, hay más probabilidad de perder paquetes por falta de potencia o por interferencias de otros dispositivos Bluetooth encendidos.

4.7.1 Test en entorno aislado

Para la realización de las capturas de este test en ningún momento se cambió ningún dispositivo de sitio, analizando así un entorno estático. Tampoco hubo movimiento de personas ni interferencias externas que puedan alterar los resultados.

En las siguientes gráficas se observa el comportamiento de la potencia en función del canal. El análisis de los dispositivos se realizó en un entorno "solitario" con ausencia de gente y cualquier otro dispositivo Bluetooth que pudiera interferir las medidas.

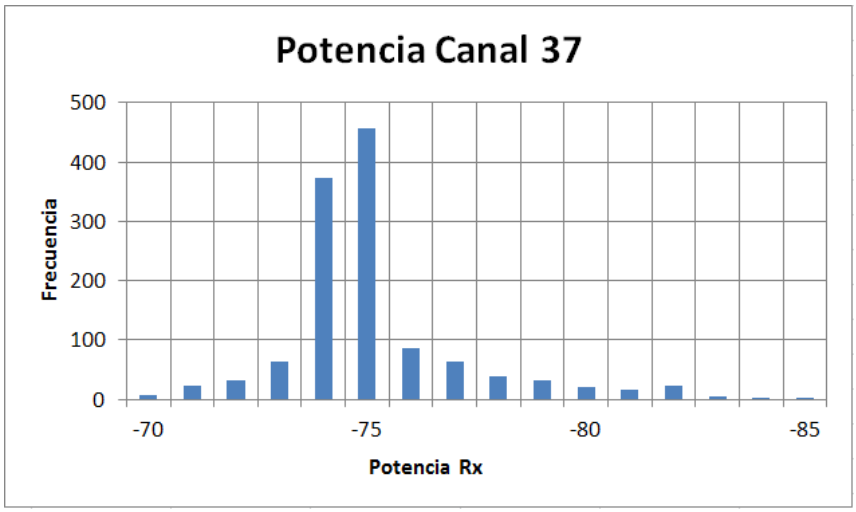

Figura 46: Frecuencia de potencias en entorno aislado (Canal 37)

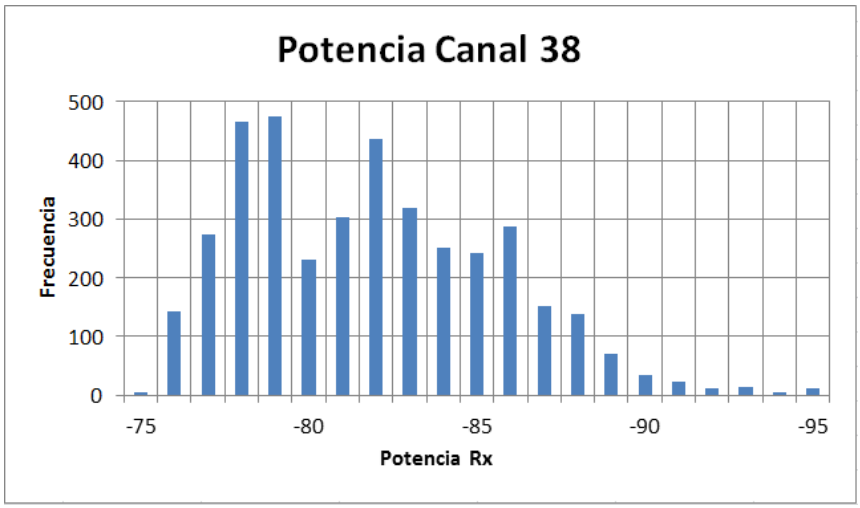

Figura 47: Frecuencia de potencias en entorno aislado (Canal 38)

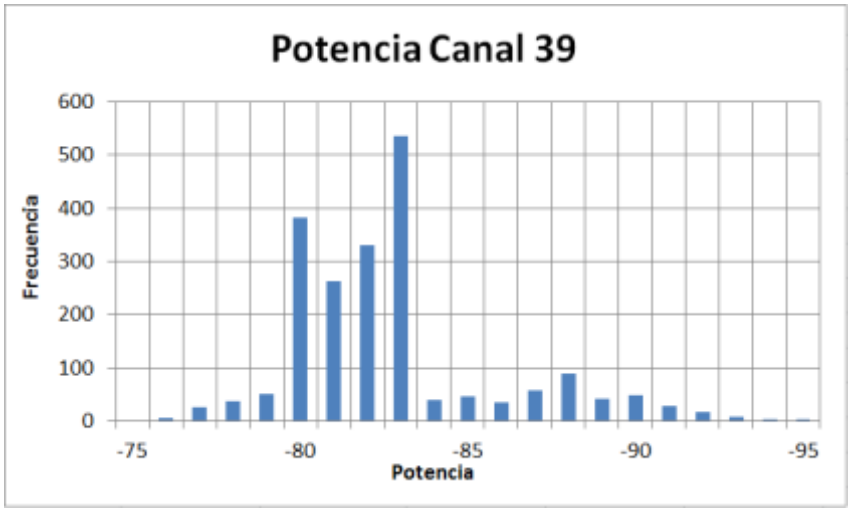

Figura 48: Frecuencia de potencias en entorno aislado (Canal 39)

Deducimos a partir de las gráficas que la potencia media recibida no es la misma para los diferentes canales. Pero sí que hay un factor común en las tres gráficas y es que las potencias recibidas se sitúan muy cerca del valor medio de cada canal. Para el caso del canal 37, la potencia media es aproximadamente de 75 dBm y el resto de potencias capturadas oscilan entre valores muy cercanos a ella, exceptuando muy pocos paquetes que sí que se alejan de la media.

4.7.2 Test con influencia de dispositivos y personas

Este test también se ha realizado con una configuración estática, dónde los dispositivos no se han movido de sitio durante las capturas. La diferencia respecto a la prueba anterior es que en este entorno sí que había gente moviéndose con total normalidad y varios dispositivos Bluetooth encendidos que no forman parte de nuestros dispositivos.

Las capturas en este entorno son diferentes a las anteriores, también se ha hecho un experimento estático, pero las condiciones de propagación no son las mismas. Las siguientes gráficas muestras la potencia recibida en los canales 37, 38 y 39.

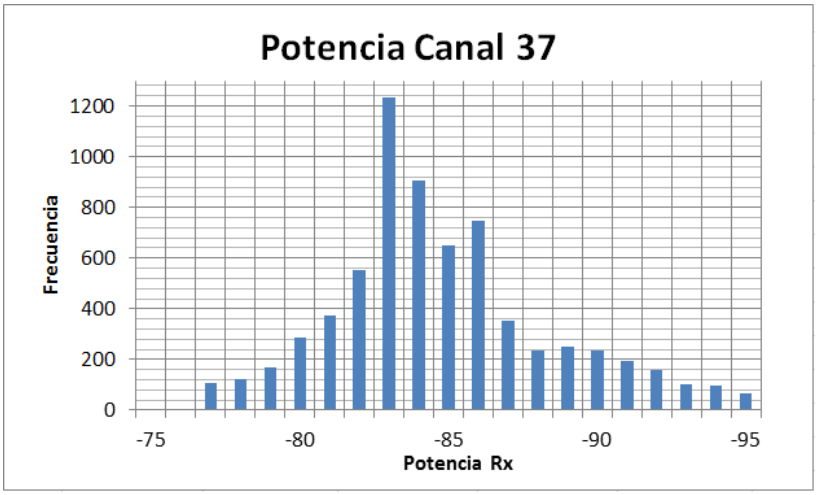

Figura 49: Frecuencia de potencias en entorno interferente (Canal 37)

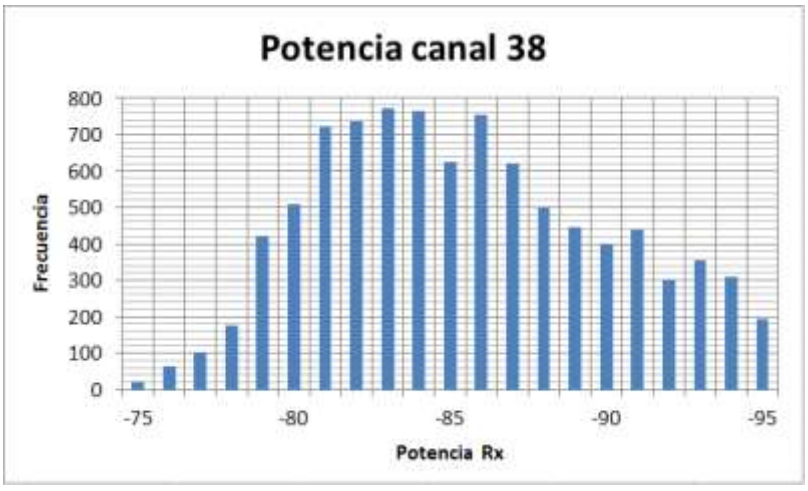

Figura 50: Frecuencia de potencias en entorno interferente (Canal 38)

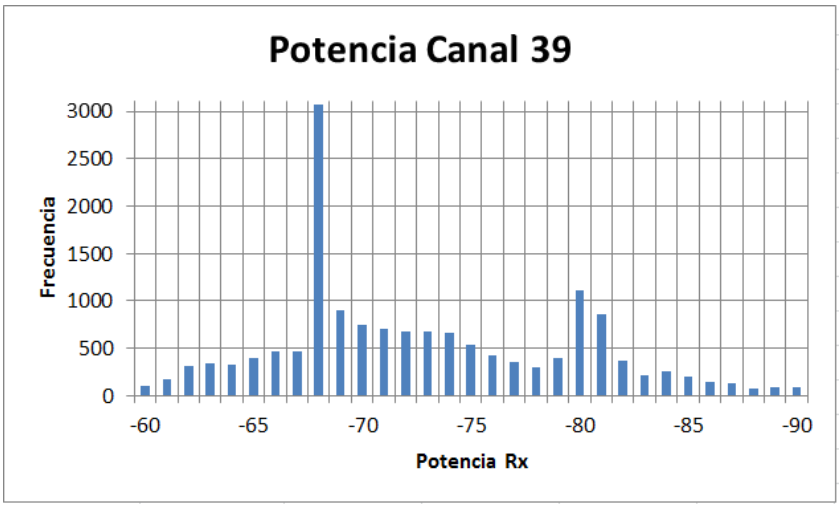

Figura 51: Frecuencia de potencias en entorno interferente (Canal 39)

Observamos de las tres figuras anteriores que la potencia captada en cada canal, al estar en un entorno variable, es más dispersa que para el caso de un entorno aislado. Esto se debe a que no podemos controlar el movimiento de la gente ni las interferencias que se producen entre los paquetes que recibe nuestro dispositivo, provocando pérdidas de estos.

Centrándonos solo en la potencia recibida en el canal 38, figuras 47 y 50, encontramos que para ambos casos ha recibido también potencias muy dispersas que se alejan del valor medio, en comparación con los otros dos canales.

Si nos centramos en la potencia recibida en cada canal, pero para los dos entornos diferentes, observamos que la potencia media recibida para los canales 37 y 38 en un entorno aislado es mucho mayor que en un entorno con factores externos. Esto se debe a se producen menos colisiones y el dispositivo es capaz de capturar más paquetes por unidad de tiempo, ya que estos no se pierden.

4.8 Análisis de la Maratón de Barcelona

El análisis de la Maratón de Barcelona es una de las partes más importantes de este trabajo, ya que se obtienen resultados muy importantes para llegar a la conclusión de si la propuesta para detectar a corredores en eventos de running sería válida.

Este estudio muestra la distribución de personas en este tipo de cursas, la cantidad de gente que se agrupa y llegan a la vez a los puntos de control, el máximo número de personas que llegaríamos a tener en nuestra zona de cobertura y también el máximo número de corredores que entrarían por segundo.

Para ello se han obtenido los datos de los corredores de la última maratón de Barcelona de una página web oficial[2]. Se han analizado a todos los corredores de esta maratón en dos puntos de control que son a 5 km y 10 km, ya que si se hubieran analizado los datos en la salida, los corredores estarían todos aglomerados.

4.8.1 Análisis a 5 km de la Maratón de Barcelona

Teniendo en cuenta las pruebas realizadas anteriormente con varias configuraciones, se acaba adoptando un radio de cobertura de 25 metros, distancia suficiente que abarca el tamaño de la calle más grande por la cual pasa el recorrido de este evento. Por lo tanto, la programación de dispositivo es similar a la tercera configuración del test en movimiento para los 52 dispositivos.

Con estos datos y los tiempos de todos los corredores se han obtenido las gráficas correspondientes al máximo número de corredores que tendríamos en cualquier momento de la cursa en nuestra zona de cobertura y las personas que entrarían por segundo.

En la figura 52 observamos que el máximo número de corredores que entrarían por segundo en nuestra zona de cobertura serían 22. Teniendo en cuenta que solo necesitamos detectar a los dispositivos de los corredores cuando se acerquen al receptor, el mejor tipo de *Advertising* que utilizaríamos sería el Nonconnectable Undirected Advertisin, ya que no necesitamos solicitar información extra de ningún corredor ni establecer conexión con ningún dispositivo.

[2] ChampionChip (2012-2017). http://www.championchip.cat/web/

Si volvemos al test Discovery Time (apartado 4.6.1) observamos que el tiempo de descubrimiento de 26 dispositivos con este tipo de *Advertisings* es de 0.3067s. Con estos datos, seríamos capaces de detectar hasta 26 dispositivos en muy poco tiempo y teniendo en cuenta que los corredores están durante unos segundos en nuestra zona de cobertura, sería posible detectarlos a todos.

Esto ocurriría solo en los picos de la gráfica, es decir, cuando llegan grandes grupos de corredores, pero la media está en que llegan aproximadamente 6 corredores por segundo, la cual cosa facilitaría el reconocimiento de estos.

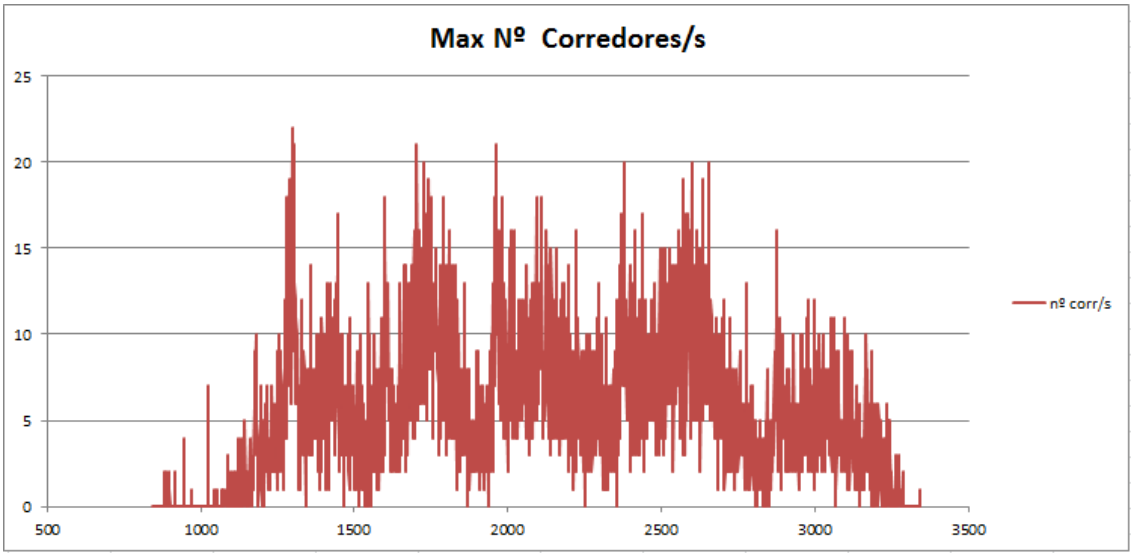

Figura 52: Máximo Nº corredores por segundo en zona cobertura [5 Km]

En la figura 53 observamos el total de dispositivos que podríamos tener en nuestra zona de cobertura enviando *Advertisings* a la vez. Este se sitúa en aproximadamente 300 corredores, motivo de mucha interferencia y pérdida de paquetes, la cual cosa complicaría la capacidad de detección de todos los corredores.

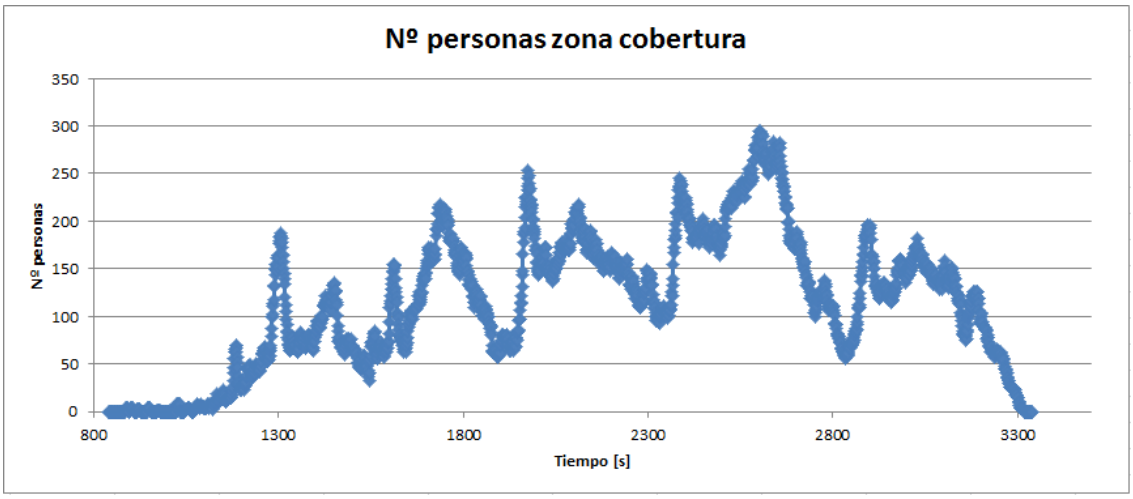

Figura 53: Máximo Nº corredores en zona de cobertura [5 Km]

Parecen demasiados corredores para detectar, pero si somos capaces de detectar en menos de un segundo a 26 corredores y unos instantes después apagamos el chip que ya haya sido detectado, conseguiríamos que éstos no provocaran interferencia y se detectarían a todos los corredores.

4.8.2 Análisis a 10 km de la Maratón de Barcelona

Para los datos de los corredores a 10 km, la configuración del dispositivo es la misma que para el caso de los 5 km.

En la figura 54 vemos que el máximo número de corredores por segundo que entrarían en nuestra zona de cobertura sería de 33 personas. Teniendo en cuenta que somos capaces de detectar a 26 dispositivos en prácticamente un tercio de segundo, detectar a 7 dispositivos más no sería un problema, ya que también conseguiríamos detectar los 33 en menos de un segundo.

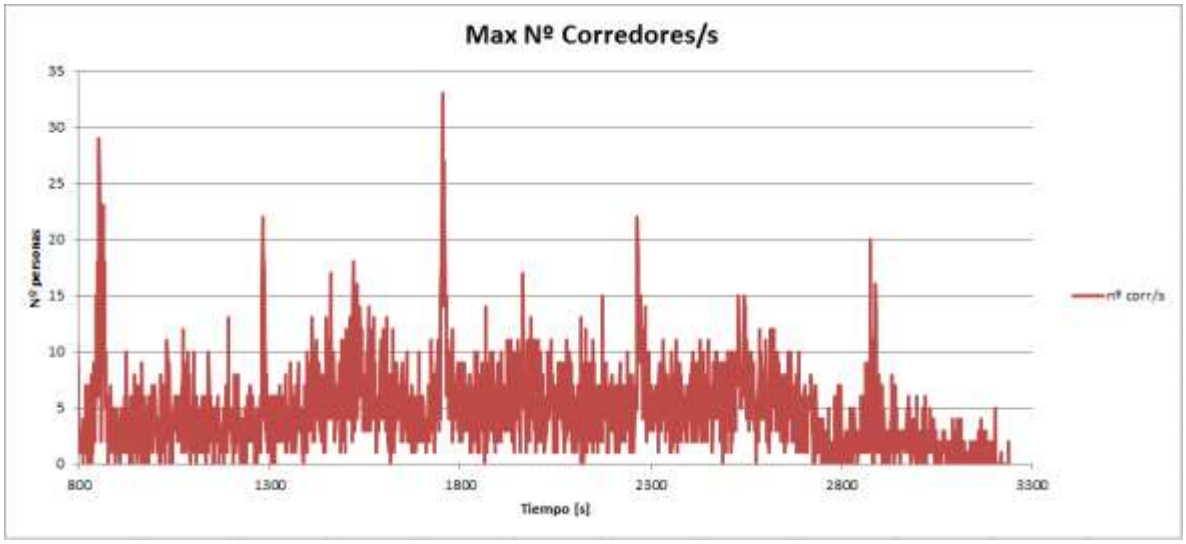

Figura 54: Máximo Nº corredores por segundo en zona cobertura [10km]

La media de corredores que llegan por segundo en este punto de la maratón sería de 9 corredores, por lo tanto, tendríamos un margen entre grupos de corredores para conseguir detectarlos a todos. Si llegan de golpe 33 corredores en un segundo, el siguiente grupo de corredores no será también de 33, sino que el número de corredores se aproximará más a la media, entonces durante ese segundo si no se han podido detectar a todos los corredores, se podrán detectar un segundo más tarde, ya que no habrá tanta gente aglomerada en el punto de control.

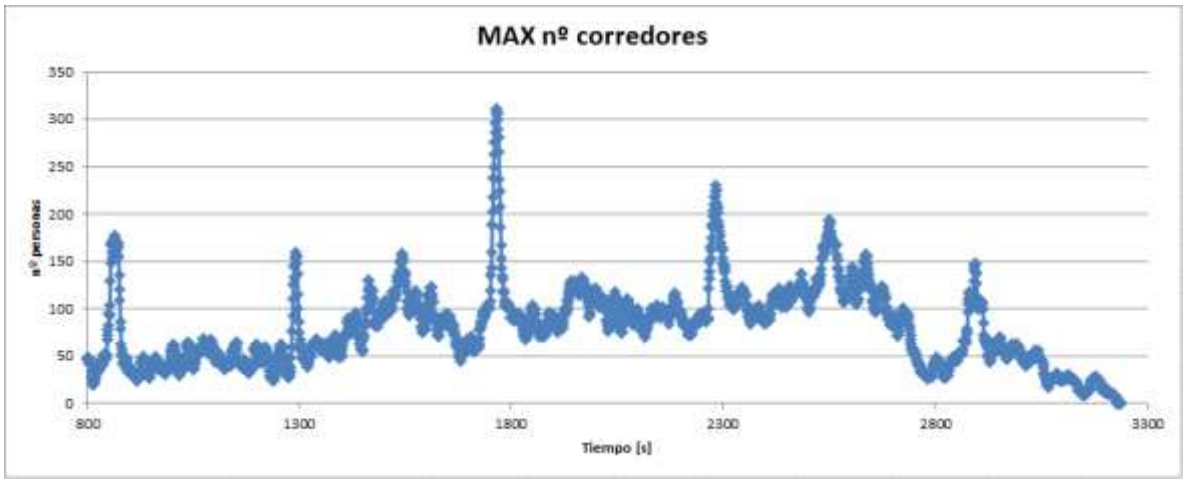

Figura 55: Máximo Nº corredores en zona de cobertura [10 Km]

En la figura 55 observamos el total de dispositivos que podríamos tener en nuestra zona de cobertura enviando *Advertisings* a la vez. También serían 300 corredores, de tal manera que adoptaríamos la misma solución de apagar los dispositivos una vez detectados y que se volvieran a encender pasado un tiempo. Con esto, tendríamos el control de la maratón.

También se ha realizado un estudio de la Salida de esta maratón, con el objetivo de ver como salen los corredores y con qué diferencia de tiempo. En la figura 56 observamos que los corredores van saliendo en grupos y separados un cierto tiempo con el grupo que viene a continuación.

Gracias a esta distribución de corredores, nos resulta más fácil detectar a los dispositivos que llevarían incorporados los corredores, ya que así van más dispersos y los grupos de personas que llegan a nuestra zona de cobertura son de menor cantidad de personas.

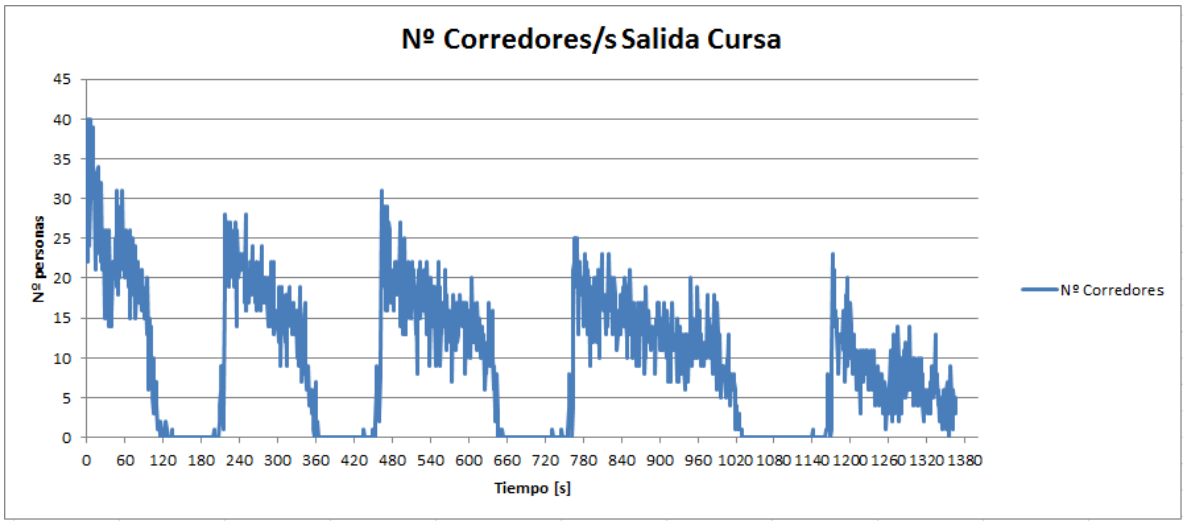

Figura 56: Nº corredores por segundo en Salida

Capítulo 5: Conclusiones y resultados

El objetivo de este proyecto es estudiar y analizar la viabilidad de un sistema BLE para la detección de deportistas en eventos con gran cantidad de personas. Para ello, después de haber presentado las características de esta tecnología se han tenido en cuenta los siguientes aspectos:

- Los procedimientos de Scanning y Advertising, dónde se explica el funcionamiento de cada uno de ellos y los parámetros que hay que tener en cuenta para su correcta configuración. Al variar un parámetro de esta configuración puede dejar de ser útil para la finalidad con la que se ha diseñado. Analizando los diferentes tipos de Advertising y los modos de Scanning, se han seleccionado los tipos adecuados para la implementación del sistema de detección de corredores.

- La caracterización del dispositivo BLE empleado obteniendo cuál es su diagrama de radiación para diferentes posiciones y también la caracterización de energía para saber su consumo con la configuración deseada.

- Test de cobertura para diferentes programaciones del dispositivo y ver el alcance para cada una de ellas, eligiendo finalmente la que más se ajusta a las características de nuestro sistema. Capturas de mensajes de Advertising para ver su funcionamiento respecto al consumo de cada dispositivo.

- Pruebas con los dispositivos en movimiento, dónde se obtiene que el alcance del dispositivo no es el mismo para un sistema estático que un sistema en el cual uno o varios dispositivos se encuentran en movimiento, indiferentemente de si se trata del transmisor o receptor.

- Medidas con decenas de dispositivos transmitiendo al mismo tiempo para calcular tiempos de descubrimiento de estos, haciendo una analogía con lo esperado en este tipo de eventos. Observar que el comportamiento del dispositivo para los diferentes canales por dónde se envían los mensajes de Advertising no es exactamente el mismo.

- Análisis de la maratón de Barcelona, dónde se han obtenido los tiempos de los corredores y la cantidad de personas que podrían llegar a nuestra zona de cobertura. Estos datos han sido cotejados con todos los resultados obtenidos en las pruebas anteriores, dando un resultado a la idea propuesta en este proyecto.

Como resultado de la investigación estadística presentada, es posible concluir que es viable conseguir mediante un receptor BLE detectar a la totalidad de los corredores en eventos con gran presencia de personas.

Desde el punto de vista económico nos encontramos delante de una tecnología de muy bajo consumo como se ha comentado durante todo el trabajo y muy utilizada cotidianamente por la gran mayoría de personas. Para llevar a cabo esta propuesta sería necesario el despliegue de dispositivos BLE en las zonas adecuadas para ello y los corredores tan solo tendrían que comprar un chip que su precio no es muy elevado.

Finalmente comentar que el objetivo del proyecto ha salido adelante y sí que es viable la posibilidad de introducir este sistema de bajo consumo en este tipo de eventos. Aunque no en todas las pruebas realizadas se han obtenido los resultados necesarios para este sistema propuesto, se han obtenido también resultados interesantes que se podrían aplicar para otro tipo de deporte.

Bibliografía

[1] Huidobro, J.M. (2003), *Tecnologías avanzadas de Telecomunicaciones*, Madrid, España. Thomson paraninfo.

[2] Catarina (2007). El estándar Bluetooth: IEEE 802.15.1. http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/archundia_p_fm/capitulo3.pdf

[3] Group, Bluetooth Special Interest. *Bluetooth Core Specification v4.2*

[4] Group, Bluetooth Special Interest. *BLUETOOTH SPECIFICA*TION Version 4.2 [Vol 2], *BR/EDR Controller Volume*

[5] Carrillo Pérez, M.S. (2006). HCI Tools. Estudio y análisis de rendimiento Bluetooth. https://www.iit.comillas.edu/pfc/resumenes/4505a8eba9cfc.pdf

[6] Red Bear Company (2015). Red Bear Lab. http://redbearlab.com/blenano/

[7] Sena Technologies. (2017). Sena networks, Estados Unidos: http://www.senanetworks.com/ud100-g03.html

[8] Trust. http://www.trust.com/es/product/18187-Bluetooth-4-0-adapter

[9] Pérez-Diaz, D., & Valenzuela, & J.L., Hernández, A. & Valdovinos, “A. Analytical and Experimental Performance Evaluation of BLE Neighbor Discovery Process Including Non-Idealities of Real Chipsets”

ANEXOS

TÍTULO DEL TFG: Análisis del sistema de comunicaciones BLE

TITULACIÓN: Grado en Ingeniería de Sistemas de Telecomunicación

AUTOR: Jorge Alcalá Colmenero

DIRECTOR: Jose Luis Valenzuela

DATA: 21 de julio del 2017

Anexos

Anexo 1: Scripts

SCAN ACTIVO

```
#!/bin/bash

#Deshabilitamos el Inquiry Scan y Page Scan
    sudo hciconfig -a hci0 noscan

#Primero deshabilitamos el Scan para poder poner los
parámetros que nos interesan
    sudo hcitool -i hci0 cmd 0x08 0x000C 00 00

#Despues ponemos los parámetros: 00 (activo= con paquetes
Scan_req), LE_Scan_Interval 0xXXXX, LE_Scan_Window 0xXXXX,
...
    sudo hcitool -i hci0 cmd 0x08 0x000B 01 A0 00 A0 00 00
00

#Por último hablitamos el scaning con los parámetros
puestos
    sudo hcitool -i hci0 cmd 0x08 0x000C 01 00
```

SCAN PASIVO

```
#!/bin/bash

#Primero deshabilitamos el Scan para poder poner los
parámetros que nos interesan
    sudo hcitool -i hci0 cmd 0x08 0x000C 00 00

#Despues ponemos los parámetros: 00 (pasivo= con paquetes
Scan_req), LE_Scan_Interval 0xXXXX, LE_Scan_Window 0xXXXX,
...
    sudo hcitool -i hci0 cmd 0x08 0x000B 00 80 00 40 00 00
00

#Por último hablitamos el scaning con los parámetros
puestos
    sudo hcitool -i hci0 cmd 0x08 0x000C 01 00
```

SCAN CONTINUO

```
#!/bin/bash

#Deshabilitamos el Inquiry Scan y Page Scan
    sudo hciconfig -a hci0 noscan

#Primero deshabilitamos el Scan para poder poner los
parámetros que nos interesan
    sudo hcitool -i hci0 cmd 0x08 0x000C 00 00

#Despues ponemos los parámetros: 00 (pasivo= sin paquetes
Scan_req), LE_Scan_Interval 0xXXXX, LE_Scan_Window 0xXXXX,
...
    sudo hcitool -i hci0 cmd 0x08 0x000B 00 20 03 20 03 00
00

#Por último hablitamos el scaning con los parámetros
puestos
    sudo hcitool -i hci0 cmd 0x08 0x000C 01 00
```

SCAN 50%

```
#!/bin/bash

#Deshabilitamos el Inquiry Scan y Page Scan
    sudo hciconfig -a hci0 noscan

#Primero deshabilitamos el Scan para poder poner los
parámetros que nos interesan
    sudo hcitool -i hci0 cmd 0x08 0x000C 00 00

#Despues ponemos los parámetros: 00 (pasivo= sin paquetes
Scan_req), LE_Scan_Interval 0xXXXX, LE_Scan_Window 0xXXXX,
...
    sudo hcitool -i hci0 cmd 0x08 0x000B 00 A0 00 50 00 00
00

#Por último hablitamos el scaning con los parámetros
puestos
    sudo hcitool -i hci0 cmd 0x08 0x000C 01 00
```

Habilitar el Advertiser

```
#!/bin/bash

# Configuracion del Advertiser

#Deshabilitamos el Inquiry Scan y Page Scan
    sudo hciconfig -a hci1 noscan

#Primero ponemos los parámetros (OGF=0x08 OCF=0x0006
Adv_Interval_Min=0xXXXX             Adv_Interval_Max=0xXXXX
Adv_Type=0x03...)
    sudo hcitool -i hci1 cmd 0x08 0x0006 A0 00 A0 00 03 00
00 00 00 00 00 00 00 07 00

#Despues ponemos los datos (OGF=0x08 OCF=0x0008 Name=09
"STOP=73 74 6F 70" Flags=01 y el relleno en ceros para
completar los 32B)
  #  sudo hcitool -i hci1 cmd 0x08 0x0008 09 05 09 73 74 6F
70 02 01 08 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00

#Habilitamos los Advertising con los parámetros previamente
puestos
    sudo hcitool -i hci1 cmd 0x08 0x000A 0x01
```